1 Introduction

Laughter is a powerful cue in communication. It communicates to listeners the emotional state of the speaker. Emotion detection is beneficial in human-machine interaction. For example, it could be used to automatically detect an opportune time for a digital camera to take a picture [1]. Laughter can also be utilized in speech processing by identifying jokes or topic changes in meetings and improve speech-to-text accuracy by recognizing non-speech sounds[2]. Furthermore, many people have distinct laughs so hearing a person’s laugh helps in the auditory recognition of others. In order to examine the many uses of laughter in the future, we first built an automatic laughter detector, which is the goal of this project.

Previous work has been done in both characterizing laughter [3] [4] [5] and building automatic laughter detectors [1] [2] [6] [7]. Bachorowski [4] found that laughter is highly variable and difficult to stereotype. Though compared to speech, laughter had more source related variability. Provine concluded that laughter is usually a series of short syllables repeated approximately every 210 ms [5].

In order to detect when multiple people laughed, Kennedy and Ellis [2] used a support vector machine (SVM) classifier trained on mel frequency cepstral coefficients (MFCCs), delta MFCCs, modulation spectrum, and spatial cues. The data was split into one second windows, which were classified as multiple speaker laughter or non-laughter events. They achieved a true positive rate of 87% and a false positive rate of 13%.

Truong and van Leeuwen [6] used gaussian mixture models (GMM) trained with perceptual linear prediction (PLP) features, pitch and energy, pitch and voicing, and modulation spectrum. The experiments were run on presegmented laughter and speech segments. Determining the start and end time of laughter was not part of the experiment. They built models for each of the four features. The model trained with PLP features performed the best, at 13.4% EER for a data set similar to the one used in our experiments.

The goal of this experiment is to automatically detect the onset and offset of laughter. In order to do so, a neural network with one hidden layer was trained with MFCC and pitch features, which will be described in Section 3.2. This task is slightly different from both Kennedy and Ellis, who used 1 second windows, and Truong and van Leeuwen, who tested on presegmented data.

In Section 2, we discuss the data that was used in this experiment. Section 3 describes our neural network. Results from the experiment are given in Section 4 and in Section 5.
we conclude and discuss our results.

2 Data

We used the ICSI\textsuperscript{1} Meeting Recorder Corpus\textsuperscript{8} to train and test the detector. It is a hand transcribed corpus of multi-party meeting recordings, in which each of the speakers wore close-talking microphones. Distant microphones were also recorded; however, they were not used in this experiment. The full text was transcribed as well as non-lexical events, including coughs, laughs, lip smacks, etc. There were a total of 75 meetings in this corpus. Similar to the work done by Kennedy and Ellis\textsuperscript{2} and Truong and van Leeuwen \textsuperscript{6}, we trained and tested on the ‘Bmr’ subset of the corpus, which included 29 meetings. The first 26 were used for training and the last 3 were used to test the detector.

In order to “clean” the data, we only trained and tested on data that was transcribed as pure laughter or pure non-laughter. Cases in which the hand transcribed documentation had both speech and laughter listed under a single start and stop time were disregarded. Furthermore, if a speaker was silent over a period of time then their channel at that time was not used in training. This reduced cross-talk (other speakers appearing on the designated speaker’s channel) and allowed us to train on channels only when they were in use. All of the data was tested; but only time that was transcribed as pure laughter or pure non-laughter was included in the computation of the equal error rate (EER). Table 1 has the statistics of the “clean” data. The average laugh duration was 1.615 seconds with a standard deviation of 1.241 seconds. Figure 1 shows the histogram of the laughter durations.

<table>
<thead>
<tr>
<th>Training Data</th>
<th>Testing Data</th>
<th>All Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pure Laughter (seconds):</td>
<td>5865.069 739.707</td>
<td>6604.776</td>
</tr>
<tr>
<td>Pure Non-Laughter (seconds):</td>
<td>90954.894 7766.517</td>
<td>98721.411</td>
</tr>
<tr>
<td>Percentage Pure Laughter (%):</td>
<td>6.058 8.696</td>
<td>6.271</td>
</tr>
</tbody>
</table>

3 Method

3.1 Neural Network

A neural network with one hidden layer was used to classify feature vectors as either laughter or non-laughter. A schematic of a neural network is shown in Figure 2. It consists of input units, hidden units, and output units\textsuperscript{9}. In this case the input units are the features and the two output units are the probability it was laughter and the probability it was not laughter. The input units are linked to each of the hidden units. The hidden units then take a weighted sum of the input units to get $b_i = \sum_j W_{i,j} F_j$, where $W_{i,j}$ is the weight associated with feature $F_j$ and hidden unit $A_i$. These weights are determined via training data. An activation function, $g$, is applied to the sum, $b_i$, to determine the value of $A_i$. Similarly, to compute the output of the neural net a weighted sum is taken of the hidden

\textsuperscript{1}International Computer Science Institute; Berkeley, CA
units and a softmax activation function is applied to the sum in order to determine the output values, the posterior probability of laughter and non-laughter.

In order to prevent over-fitting, the data used to train the neural network was split into two groups, training (the first 21 Bmr meetings) and cross validation (the rest of the original training set). The weights are adjusted based on the first 21 Bmr meetings via the back-propagation algorithm, which modifies the weight based on the partial derivative of the error with respect to each weight. After each epoch the cross validation frame accuracy (CVFA) is evaluated. The CVFA is the ratio of true negatives and true positives to all cross validation data. Using the system used at ICSI, the learning rate was initially set to 0.008. Once the CVFA does not increase by at least 0.5% from the previous epoch, the learning rate is halved at the beginning of subsequent epochs. The next time the CVFA does not improve by 0.5% training is stopped.

Figure 1: Histogram of Laugh Duration

Figure 2: A neural network with n input units, 200, hidden units, and 2 output units

Figure 2: A neural network with n input units, 200, hidden units, and 2 output units
3.2 Features

3.2.1 Mel Frequency Cepstral Coefficients

Mel frequency cepstral coefficients (MFCCs) are coefficients obtained by taking the fourier transform of a signal, converting it to the mel scale, and finally taking the discrete cosine transform of the mel scaled fourier transform \[10\]. The mel scale is a perceptual scale used to more accurately portray what humans hear. In this experiment, MFCCs were used to capture the spectral features of laughter and non-laughter. The first order regression coefficients of the MFCCs (delta MFCCs) and the second order regression coefficients (delta-delta MFCCs) were also computed and used as features for the neural network. We used the first 12 MFCCs (including the energy component), which were computed over a 25 ms window with a 10 ms forward shift, as features for the neural network. MFCC features were extracted using HTK.

3.2.2 Pitch

From laughter characterization papers, it was concluded that pitch features were different for laughter than for speech. In \[3\], they found that the fundamental frequency ($F_0$) pattern of laughter did not decline as it typically does in speech. In \[4\], they found $F_0$ to have large ranges during laughter.

Using the ESPS pitch tracker get_f0 \[11\], we extracted the $F_0$, rms value, and ac peak (the highest normalized cross correlation value found to determine $F_0$) for each frame. The delta and delta-delta coefficients were computed for each of these features as well.

4 Experiments

Since the frames are so short in duration and in this data set people generally laughed for a consecutive 1.615 seconds at a time, we realized we should feed multiple frames into the neural net to determine whether or not a person is laughing. After trying 0.25, 0.50, 0.75, and 1 second windows we found that a window of 0.75 seconds worked best, based on the CVFA. We wanted the classification of laughter to be based on the middle frame so we set the offset to be 37 frames.

We also had to determine the number of hidden units. Since MFCCs were the most valuable features for Kennedy and Ellis\[2\], we used MFCCs as the input units and modified the number of hidden units to be 50, 100, 200, and 300 while keeping all other parameters the same. Based on the accuracy on the cross validation set, we saw that 200 hidden units performed best. The other parameters used for the neural network, including the learning rate, were set to values which work well for speech recognition at ICSI \[12\].

As stated earlier, the neural network was trained only on data that was included in the transcript as either pure laughter or pure non-laughter. After the weights were set based on this training data, all of the test data was run through the neural network and given two output scores. The first output score was the probability the frame was non-laughter and the second output score was the probability the frame was laughter. Although the entire test set was given two output scores, only the pure laughter and pure non-laughter frames
were used to compute the detection error trade-off (DET) curves and the equal error rate (EER).

The neural networks were first trained with each of the features individually (i.e. MFCCs, delta MFCCs, delta-delta MFCCs, \(F_0\), delta \(F_0\), etc). Then we combined each class of features (MFCCs, \(F_0\), rms value, ac peak) with their respective deltas and delta-deltas and retrained the neural networks. The EERs are shown in Table 2 and the DET curves are shown in Figures 3, 4, 5, and 6. Each figure contains four DET curves: the feature itself, delta-feature (* D), delta-delta feature (* A), and the combination of the feature, delta, and delta-delta (* ALL).

Figure 3: DET curve for MFCC features

Figure 4: DET curve for ac peak features
Figure 5: DET curve for F0 features

Figure 6: DET curve for rms features

| Table 2: Equal Error Rates (%) |
|-------------------|---|---|---|---|
| MFCCs | $F_0$ | RMS | AC Peak |
| Feature      | 11.35  | 23.26  | 32.22  | 16.75  |
| Delta        | 9.62   | 24.42  | 26.52  | 22.37  |
| Delta-Delta  | 11.23  | 27.83  | 26.62  | 27.61  |
| All          | 10.66  | 22.80  | 26.01  | 16.72  |
5 Discussion and Conclusions

From Table 2, it is clear that MFCC features outperformed pitch related features. This is consistent with Kennedy and Ellis’ [2] results. For Truong and van Leeuwen [6], PLP features outperformed the other features. PLP, like MFCCs, are perceptually scaled spectrums so it is not surprising that they performed well for the task of laughter detection too.

AC peak values also performed well, which suggests that the cross correlation of an audio signal helps in detecting laughter. This seems reasonable since laughter is repetitive [5]. However Provine [5] found that the repetitions were 210 ms apart, which exceeds the time used to compute the correlation.

In the future, we plan to combine the features both on the score level and the feature level. The added features would likely improve our results.

Additional features could be utilized to detect laughter. Trouvain noted the repetition of a consonant-vowel syllable structure [13]. Based on this model, we could run a phoneme recognizer on the audio and detect when a phoneme is repeated. Another approach would be to compute the modulation spectrum. This should also portray the repetitive nature of laughter.

In conclusion, we have shown that neural networks can be used to automatically detect the onset and offset of laughter with an EER of 9.62%. These results are slightly better than previous experiments which classified presegmented data. By adding more features and combining features we hope to further improve these results.

6 Acknowledgements

We would like to thank Nikki, Chuck, Howard, Lara, Khiet, Mathew, and Galen for their help with this experiment.
References


[10] “Mel frequency cepstral coefficient,”

    August 1993.
