Improving Emotion Detection with Sub-clip Boosting

Ermal Toto, Brendan J. Foley, and Elke A. Rundensteiner

Worcester Polytechnic Institute, Worcester, MA, USA
\{toto,bjfoley,rundenst\}@wpi.edu

Abstract. With the emergence of systems such as Amazon Echo, Google Home, and Siri, voice has become a prevalent mode for humans to interact with machines. Emotion detection from voice promises to transform a wide range of applications, from adding emotional-awareness to voice assistants, to creating more sensitive robotic helpers for the elderly. Unfortunately, due to individual differences, emotion expression varies dramatically, making it a challenging problem. To tackle this challenge, we introduce the Sub-Clip Classification Boosting (SCB) Framework, a multi-step methodology for emotion detection from non-textual features of audio clips. SCB features a highly-effective sub-clip boosting methodology for classification that, unlike traditional boosting using feature subsets, instead works at the sub-instance level. Multiple sub-instance classifications increase the likelihood that an emotion cue will be found within a voice clip, even if its location varies between speakers. First, each parent voice clip is decomposed into overlapping sub-clips. Each sub-clip is then independently classified. Further, the Emotion Strength of the sub-classifications is scored to form a sub-classification and strength pair. Finally we design a FilterBoost-inspired “Oracle”, that utilizes sub-classification and Emotion Strength pairs to determine the parent clip classification. To tune the classification performance, we explore the relationships between sub-clip properties, such as length and overlap. Evaluation on 3 prominent benchmark datasets demonstrates that our SCB method consistently outperforms all state-of-the-art methods across diverse languages and speakers.

1 Introduction

Motivation. In recent years voice activated assistants such as Amazon Echo\(^1\), Google Home\(^2\) and Siri\(^3\) have become a prevalent mode of human-machine interaction. While prior findings \cite{20} indicate that human-machine interactions are inherently social, current devices lack any empathic ability. If only we could automatically extract and then appropriately respond to the emotion of the speaker, this would empower a large array of advanced human-centric applications, from customer service \cite{30} and education \cite{19}, to health care \cite{24}. As it

\(^1\) https://www.amazon.com/
\(^2\) https://madeby.google.com/home/
\(^3\) http://www.apple.com/ios/siri/
is best expressed by the advertising industry, emotion detection is key to “Capturing the heart”\textsuperscript{23} of users and customers. Thus, as machines grow more complex, emotions can no longer be ignored.

**Challenges.** Emotion detection from voice continues to be a challenging problem and thus an active area of study\textsuperscript{18,35,13,34,13,12}. To emphasize the challenges of this task, it has been observed that even humans experience difficulty in distinguishing reliably between certain classes of emotion - especially for speakers from other nationalities\textsuperscript{8}.

In addition, the location of emotion cues can vary both between and within a voice clip. That is, in voice clips of different content, cues will naturally be in different locations. However, even for clips with the same textual content, cues will vary due to stark differences in expressing emotion between individuals\textsuperscript{2}.

**Limitations of the State-of-the-Art.** Emotion detection is an active area of research with studies ranging from text analysis\textsuperscript{36} to facial pattern recognition\textsuperscript{11}. Sentiment analysis from text suffers from problems specific to each language, even though humans share a set of universal emotions\textsuperscript{1}. In addition, even when controlling for differences in language, emotion expression is highly non-verbal\textsuperscript{15}. Thus, methods that consider facial expressions in addition to voice are at times utilized\textsuperscript{11}. However, such multi-modal data types are in many cases either too obtrusive, too costly or simply not available.

To avoid the above pitfalls, our work concentrates in emotion detection from non-verbal voice features. In this context, despite the existence of universal emotions\textsuperscript{6}, state-of-the-art methods for emotion recognition still do not generalize well between benchmark datasets\textsuperscript{7} (Table 3). Some approaches\textsuperscript{16,17} attempt to address the problem of generalization by eliminating clip outliers, and thus do not succeed to classify all data instances.

Simple multi-class regression methods have been widely used, but fail to achieve practical accuracy\textsuperscript{31,25}. Core work in emotion detection from voice uses a mixture of GMM (Gaussian Mixture Models)\textsuperscript{14}, and HMM (Hidden Markov Models)\textsuperscript{5}. Most recently, with the increased popularity of deep learning (DL), we see the rise of attempts to apply DL to this problem, in particular, Random Deep Belief Networks\textsuperscript{35}, Recurrent Neural Networks (RNN)\textsuperscript{13}, and Convolutional Neural Networks (CNN)\textsuperscript{34,3,12}. Convolutional Neural Networks (CNN)\textsuperscript{34,3,12} achieve better results than other state-of-the-art methods, but still fail to generalize to all benchmark datasets and leave room for improve-
ment. Due to these shortcomings, more work is needed in voice-based emotion recognition.

**Our Proposed Approach.** To solve this problem, we set out to design a novel Sub-clip Classification Boosting methodology (SCB). SCB is guided by our hypothesis that emotion tends not to be sustained over the entire length of the clip – not even for clips that have been carefully created for a benchmark. Therefore, the prediction of the emotion state of a clip needs to carefully consider its sub-clips. Sub-clipping (Section 5.1) effectively works as a moving lens (Figure 1) over the sound signal - providing the first step to solving the undetermined boundary problem observed in a prior survey [2].

Next, a high dimensional non-verbal feature extraction from these voice sub-clips is tamed by prioritized feature selection [21] to reduce data dimensionality from more than 1500 features to a few hundred. As we will demonstrate, a dataset independent feature selection approach guided by the emotion classification problem itself, achieves better generalization across diverse benchmarks and emotion classes [4], which is one of the challenges observed with current state-of-the-art. In addition, utilizing non-verbal features overcomes classification challenges due to different languages.

To account for the unequal distribution of emotion cues between sub-clips, an Emotion Strength (ES) score is assigned to each sub-clip classification (SC), forming a SC, ES pair. Finally, inspired in part by FilterBoost [4], SCB implements an “Oracle” that utilizes all the Sub-Classification (SC), ES pairs (of a voice clip) to determine the parent clip classification.

To evaluate SCB, we have conducted an extensive study using three popular and widely used benchmark datasets described in Section 3. The results demonstrate that compared to the state-of-art algorithms, SCB consistently boosts the accuracy of predictions in all these benchmark datasets. For repeatability [4], we have posted our SCB code base, associated data, and meta-data. To summarize, our main contributions include:

- We design the SCB framework for the robust detection of emotion classes from voice across speakers, and languages.
- We design an Oracle-based algorithm inspired by FilterBoost that utilizes sub-classification (SC), Emotion Strength (ES) pairs to classify the parent clip.
- We tune our models, and explore the relationships between sub-clip properties, such as length and overlap, and the classification accuracy.
- Lastly, we evaluate the resulting SCB framework using 3 benchmark datasets and demonstrate that SCB consistently achieves superior classification accuracy improvements over all state-of-the-art algorithms.

The rest of the paper is organized as follows. Section 2 situates our work in the context of Affective Computing, current methodologies for Emotion Detection, and Boosting Strategies. Section 3 introduces preliminary notation, and our problem definition. Section 4 introduces the Sub-clip Classification Boosting (SCB) Framework. Section 5 explains the SCB method and its sub-

4 http://solar-10.wpi.edu/toto/EMOTIVOClean.git
components. Section 6 describes our evaluation methodology and experimental results on benchmarks, while Section 7 concludes our work.

2 Related Work

**Affective Computing.** Affective computing [22] refers to computing that relates to, arises from, or deliberately influences emotion. Given that emotion is fundamental to human experience, it influences everyday tasks from learning, communication, to rational decision-making. Affective computing research thus develops technologies that consider the role of emotion to address human needs. This promises to enable a large array of services, such as better customer experience [30], intelligent tutoring agents [19] and better health care services [24]. Recent related research in psychology [27] points to the existence of universal basic emotions shared throughout the world. In particular it has been shown that at least six emotions are universal [5], namely: Disgust, Sadness, Happiness, Fear, Anger, and Surprise.

**Emotion Detection from Voice.** Detection of universal emotions from voice continues to be a challenging problem. Current state-of-the-art methods for emotion recognition do not generalize well from one benchmark database [7] to the next, as summarized in Table 3. Some approaches [16,17] attempt to address the problem of generalization by eliminating clip outliers belonging to particular emotion classes, and thus providing an incomplete solution.

SVM (Support Vector Machines) have been identified as an effective method, especially when working with smaller datasets [31]. However their accuracy remains low - not sufficient for practical purposes (Table 3). Softmax regression [25] is similar to SVM, in that it is a multi-class regression that determines the probability of each emotion class. The resulting, Softmax accuracy is higher than SVM, but still lower than the more recent state-of-the-art methods described below.

GMM (Gaussian Mixture Models) [14] are better at identifying individual differences through the discovery of sub-populations using normally distributed features.

HMM (Hidden Markov Models) [5] are a popular method utilized for voice processing, that relies on building up a probabilistic prediction from very small sub-clips of a voice clip. In contrast to SCB, the sub-clips considered by HMM are too small to contain information that can be accurately classified on its own.

More recently, with the increase in popularity of Deep Learning, Random Deep Belief Networks [35], Recurrent Neural Networks (RNN) [13], and Convolutional Neural Networks (CNN) [34,3,12] have been used for the emotion classification task. RNN [13] and Random Deep Belief Networks [35] have been shown to have low accuracy for this task (see also our Table 3). CNN [34,3,12] on the other hand has achieved some of the higher accuracy predictions for specific benchmark datasets. However, this method fails to generalize to other
sets \[34,3,12\]. This drawback may come in part from the lack of large available datasets with labeled human emotions, which are hard to obtain given the unique nature of this task. As our in-depth analysis reveals, the accuracy of CNN is comparable to previously discussed and less complex methods, while still leaving room for improvement.

Boosting Strategies. SCB is a meta-model approach, such as Boosting, Bagging, and Stacking \[26\] more specifically a type of Boosting. Boosting methods have been shown to be effective in increasing the accuracy of machine learning models. Unlike other boosting methods that work with subsets of features or data, SCB utilizes sub-instances (sub-clips) to boost the classification accuracy. Results of a direct comparison with current meta-models are given in Table 2.

3 Preliminaries

We work with three benchmark datasets for emotion detection on voice clips, namely: RML Emotion Database\[5\] [33], an emotion dataset, containing 710 audio-visual clips by 8 actors in 6 languages, displaying 6 universal \[6\] emotions, namely: Disgust, Sadness, Happiness, Fear, Anger, and Surprise. Berlin Emotion Database\[6\] [32], a German emotion dataset, containing 800 audio clips by 10 actors (5 male, 5 female), displaying 6 universal emotions and Neutral. SAVEE\[7\] [10], an English emotion dataset, containing 480 audio-visual clips by 4 subjects, displaying 6 universal emotions and Neutral. Table 1 explicitly highlights key notation used in the paper.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>(V)</td>
<td>An ordered set of voice clips.</td>
</tr>
<tr>
<td>(v_i)</td>
<td>The (i^{th}) voice clip. To distinguish between entities with known labels (current data) and new data to be classified, we use the prime notation (E.g. (v'_i) is a new voice clip.</td>
</tr>
<tr>
<td>(</td>
<td>v_i</td>
</tr>
<tr>
<td>(N)</td>
<td>Number of sub-clips.</td>
</tr>
<tr>
<td>(\Omega)</td>
<td>ratio of overlap between adjacent sub-clips.</td>
</tr>
<tr>
<td>(v_{i,j}^{N,\Omega})</td>
<td>The (j^{th}) sub-clip of (v_i), given (N) and (\Omega).</td>
</tr>
<tr>
<td>(e_i)</td>
<td>Emotion label of (v_i) and all sub-clips of (v_i).</td>
</tr>
<tr>
<td>(\hat{e}_i)</td>
<td>The predicted emotion class of sub-clip (v_{i,j}^{N,\Omega}).</td>
</tr>
<tr>
<td>(f_{i,j}^{N,\Omega})</td>
<td>Features vector extracted from (v_{i,j}^{N,\Omega}).</td>
</tr>
<tr>
<td>(\hat{e}_{i,j}^{N,\Omega})</td>
<td>Ordered set of sub-classification labels for all sub-clips of (v_i).</td>
</tr>
<tr>
<td>(W_{i,j}^{N,\Omega})</td>
<td>Ordered set of Emotion Strength scores for all sub-clips of (v_i).</td>
</tr>
</tbody>
</table>

Table 1: Preliminary Notation.

5 http://www.rml.ryerson.ca/rml-emotion-database.html
6 http://www.emodb.bilderbar.info/
7 http://kahlan.eps.surrey.ac.uk/savee/
**Problem Definition.** Given a set of voice clips $v_i' \in V$ with unknown emotion states, the problem is to predict the class $e_i'$ for each instance $v_i'$ with high accuracy.

4 The SCB Framework

To tackle the above emotion detection problem, we introduce the SCB framework (Sub-clip Classification Boosting) as shown in Figure 2. Below we describe the main processes of the SCB framework.

Sub-clip Generation module takes as input a voice clip, the desired number of sub-clips $N$, and a percent overlap parameter $\Omega$, to generate $N$ sub-clips of equal length. During model training sub-clips are labeled after their parent clip. This process is further described in Section 5.1.

Feature Extraction is performed using openSMILE\(^8\) which generates a set of over 1500 features. Due to the high dimensionality of the data generated by openSMILE, at model training a feature selection mechanism is applied to reduce the complexity of further steps – allowing us to instead focus on the core classification technology.

Feature Selection is applied during model training. For this task we explored several strategies for dimension selection, all equally applicable. Finally, in our experiments, we work with one found to be particularly effective in our context, namely, the maximization of the mean decrease in node impurity based on the Gini Index for Random Forests\(^{21}\). Based on this metric, a vector with the importance of each attribute is produced. Guided by this importance criteria, the system selects a set of top

---

\(^8\) http://audeering.com/technology/opensmile/
Improving Emotion Detection

features. For the purposes of this research, we select the top 350 features, a more than 4 fold reduction from the original set of features. However, our approach is general, and alternate feature selection strategies could equally be plugged into this framework.

**Cached Feature Selection (On-line)** During model deployment the cached list of features selected during training is looked-up to reduce the complexity of Feature Extraction from new clips.

**Sub-Clip Classification** module trains sub-classification models to assign (universal and neutral) emotion labels to future sub-clips. Several ML algorithms are evaluated for this task (Section 5.2).

**Sub-Clip Emotion Strength Scores** are computed for each sub-classification. To reduce bias, SCB utilizes a non-linear ensemble of multiple regression models that generate scores between 0 and 1, where a score close to 0 indicates low Emotion Strength or a weak sub-classification, while a score closer to 1 indicates high Emotion Strength or a strong sub-classification (Section 5.2).

**Clip Classification ”Oracle”** evaluates SC, and ES pairs to classify the parent clip. Several “Oracle” implementations are evaluated, namely, Cumulative Emotion Strength, Maximum Emotion Strength, and Sub-clip Voting. These are discussed in more detail in Section 5.3.

5 Sub-clip Classification Boosting

Fig. 3: Overview of SCB. Each voice clip is assigned a sub-classification, Emotion Strength pair. An “Oracle” uses these pairs to classify the parent clip.

Our proposed Sub-clip Classification Boosting (SCB) method (Algorithm 1) is based on the intuition that strong emotion cues are concentrated unevenly within a voice clip. Thus, our algorithm works as a moving lens to find sub-clips of a data instance that can be used to boost the classification accuracy of the whole. Each sub-clip is independently classified, and for each sub-classification (SC) an Emotion Strength score (ES) is also predicted. To combined SC, ES pairs into the parent clip classification, SCB implements a FilterBoost [4] inspired “Oracle” (Section 5.3).

To illustrate, assume the voice clip in Figure 3 is the input to SCB. In Line 1 of Algorithm 1, the inputs are: a voice clip $v_i'$, the number of sub-clips $N$ (in Figure 3 $N = 3$), sub-clip overlap $\Omega$, and a pre-selected set of features $F$. The length of sub-clips is than determined in Line 2 Next, the parent voice clip $v_i'$ is cut into sub-clips (Line 7) which are processed one at a time.
Algorithm 1 The SCB Method

1: input: \( v_i, N, \Omega, \mathcal{F} \)
2: SubClipLength \( \leftarrow \frac{|v_i|}{N - (N - 1) \times \Omega} \)
3: \( j \leftarrow 0 \)
4: \( \text{Start} \leftarrow 0 \)
5: \( \text{End} \leftarrow \text{SubClipLength} \)
6: while \( j < N \) do
7: \( v^{N,\Omega}_{i,j} \leftarrow \text{Subclip}(v_i, \text{Start}, \text{End}) \)
8: \( \text{Start} \leftarrow \text{End} - \Omega \times \text{SubClipLength} \)
9: \( \text{End} \leftarrow \text{Start} + \text{SubClipLength} \)
10: \( f^{N,\Omega}_{i,j} \leftarrow \text{openSMILE}(v^{N,\Omega}_{i,j}, \mathcal{F}) \)
11: \( \hat{e}^{N,\Omega}_{i,j} \leftarrow \text{C}(f^{N,\Omega}_{i,j}) \)
12: \( \hat{w}^{N,\Omega}_{i,j} \leftarrow \text{R}(f^{N,\Omega}_{i,j}, \hat{e}^{N,\Omega}_{i,j}) \)
13: \( \hat{E}^{N,\Omega}_i.\text{push}(\hat{e}^{N,\Omega}_{i,j}) \)
14: \( \hat{W}^{N,\Omega}_i.\text{push}(\hat{w}^{N,\Omega}_{i,j}) \)
15: \( j \leftarrow j + 1 \)
16: end while
17: \( \hat{e}_i = \text{Oracle}(\hat{E}^{N,\Omega}_i, \hat{W}^{N,\Omega}_i) \)
18: return \( \hat{e}_i \)

Then, openSMILE (Line 10) extracts a preselected set of features \( \mathcal{F} \), as described in Section 4, from each individual sub-clip. These features are stored as a vector \( f^{N,\Omega}_{i,j} \). In Line 11, a previously trained sub-classification model \( C \) (Section 5.2) labels the sub-clip based on its feature vector \( f^{N,\Omega}_{i,j} \). In Line 13, the ES score is computed by a regression model \( R \) (Section 5.2). Thus, each sub-clip receives a SC,ES pair of a label and score. In the example in Figure 3, the pairs are (Happy,0.9), (Neutral,0.5), and (Sad,0.1). SC labels, and ES scores are respectively saved in the \( \hat{E}^{N,\Omega}_i \) and \( \hat{W}^{N,\Omega}_i \) ordered sets.

Finally, the Classification “Oracle” evaluates the \( \hat{E}^{N,\Omega}_i \) and \( \hat{W}^{N,\Omega}_i \) ordered sets, and outputs the parent clip classification (Line 17) as described in in Section 5.3.

5.1 Sub-clip Generation

To generate sub-clips, each voice clip \( v_i \) is split into \( N \) sub-clips \( v^{N,\Omega}_{i,j} \) with \( \Omega \) overlap, where \( 0 \leq \Omega < 1 \) is relative to the length of the sub-clips. The sub-clip length \( |v^{N,\Omega}_{i,j}| \) is computed by Equation 1. Sub-clips are cut from their parent clip as described in Algorithm 1 (Lines 2-9) and illustrated in Figure 3.

\[
|v^{N,\Omega}_{i,j}| = \frac{|v_i|}{(N - (N - 1) \times \Omega)}. \tag{1}
\]

Tuning the number of clips and their overlap has interesting implications for the accuracy of the parent clip classification (Sections 6.5.2 and 6.5.1). Increasing the number of sub-clips, increases the number of independent classifications.
per clip, however it also reduces the amount of information per sub-clip. To mitigate the information reduction due to sub-clipping, sub-clip overlap adds additional, but redundant information to each sub-clip. This redundancy needs to be controlled during model training and testing in order to avoid over-fitting. Our results show that a certain amount of overlap increases parent-clip classification accuracy. By tuning these parameters we discovered optimal zones for both $N$ and $\Omega$, making these hyper parameters important to model tuning.

5.2 Sub-classification and Emotion Strength Pairs

Sub-classification (SC). A sub-clip classification Model $C$ is trained using labeled feature vectors of audio sub-clips, so that it can predict the label of a new instance represented by a feature vector $f_{i,j}^{N,\Omega}$. Therefore, this process can be expressed as $\hat{e}_{i,j}^{N,\Omega} = C(f_{i,j}^{N,\Omega})$, where $e_{i,j}^{N,\Omega}$ is the SC label of the sub-clip $v_{i,j}^{N,\Omega}$. To implement these models, we evaluate several machine learning algorithms suitable for classification, such as SVM, RandomForest, kNN, and naiveBayes.

Emotion Strength (ES). Given a predicted SC label $\hat{e}_{i,j}^{N,\Omega}$, the ES score $0 \leq \hat{w}_{i,j}^{N,\Omega} \leq 1$ of the sub-clip $v_{i,j}^{N,\Omega}$ is predicted by a previously trained regression model such that $\hat{w}_{i,j}^{N,\Omega} = R(f_{i,j}^{N,\Omega} | \hat{e}_{i,j}^{N,\Omega})$. Since SC labels are unknown prior to sub-clip classification, separate ES regression models need to be trained for each of the 7 emotions (6 universal emotions, and Neutral) that are present in the datasets. During training, categorical labels are converted to binary values, with 1 indicating the presence of the emotion for which the regression model is being trained to generate an ES score, and 0 for all other emotions. Several machine learning algorithms are tested for this task, such as SVM, KNN, ANN, and Linear Models. A non-linear ensemble of models such as $\hat{w}_{i,j}^{N,\Omega} = R_{\text{SVM}}(f_{i,j}^{N,\Omega} | \hat{e}_{i,j}^{N,\Omega}) \times R_{\text{kNN}}(f_{i,j}^{N,\Omega} | \hat{e}_{i,j}^{N,\Omega})$, was found optimal for reducing bias. Intuitively, ES scores closer to 1 are viewed as strong support for the SC label of the (SC,ES) pair, while low ES scores indicate low support for the pair’s SC label.

5.3 Clip Classification “Oracle”

We evaluate three “Oracle” implementations for the classification of the parent clips from SC, ES pairs, namely, Cumulative Emotion Strength Sub-clip Classification, Maximum Sub-clip Emotion Strength Classification and Sub-clip Voting Classification. In each of the implementations, the “Oracle” receives $\hat{e}_{i}^{N,\Omega}$ and $\hat{w}_{i}^{N,\Omega}$ ordered sets containing SC labels and ES scores, and makes a decision on the classification of the parent clip as described below.

5.3.1 Cumulative Emotion Strength (Equation 2) is an “Oracle” implementation based on the intuition that all sub-clips have a weight in the final clip classification. For example, if a parent clip is divided into 3 sub-clips, with the
following SC, ES pairs: (Fear, 0.7), (Happiness, 0.4), (Happiness, 0.6), the emotion with the highest cumulative ES (0.4 + 0.6 = 1) is Happiness.

\[ \hat{e}'_i = \arg\max_{e \in \mathcal{E}} \sum_{\forall j | \hat{e}'_{i,j}^{N,\Omega} = e} \hat{w}_{i,j}^{N,\Omega}. \]  

(2)

ES scores play no role in sub-classification, and are only used during the parent clip classification. In case of a tie, one of multiple possible labels is chosen at random.

5.3.2 Maximum Emotion Strength (Equation 3) is an “Oracle” implementation based on the intuition that certain sub-clips best represent the overall emotion of the parent clip. Therefore, this method promotes the sub-classification with the highest ES score. To illustrate, if a target clip is divided into 3 sub-clips, with the following SC, ES pairs: (Fear, 0.7), (Happiness, 0.4), (Happiness, 0.6), the parent clip would be classified as Fear based on the maximum emotion strength. In case of a tie, one of multiple possible labels is chosen at random.

\[ \hat{e}'_i = \arg\max_{e \in \mathcal{E}} (\max(\hat{w}_{i,j}^{N,\Omega} | \hat{e}'_{i,j}^{N,\Omega} = e)). \]  

(3)

5.3.3 Sub-clip Voting Classification (Equation 4) is an “Oracle” implementation based on the intuition that all sub-classifications contribute equally to the parent classification, in essence a voting ensemble. To illustrate, if a parent clip is divided into 5 sub-clips, of which three are classified as Fear one as Happiness and one as Anger, the clip will be classified as Fear. In case of a tie, one of multiple possible labels is chosen at random.

\[ \hat{e}'_i = \arg\max_{e \in \mathcal{E}} (\text{count}(\hat{e}'_{i,j}^{N,\Omega} = e)). \]  

(4)

6 Experimental Evaluation

To evaluate the performance of our SCB framework we conducted comprehensive experiments using three benchmark datasets that are commonly used to compare state-of-the-art methods. The experimental results show that SCB can achieve up to 4% improvement in classification accuracy over current state-of-the-art methods, while consistently achieving better generalization between datasets.

6.1 Benchmark Data Sets

SCB was evaluated with three popular benchmark datasets for emotion detection on voice clips described in Section 3. These datasets are chosen for their complete coverage of the six universal emotions, namely: Disgust, Sadness, Happiness, Fear, Anger, and Surprise. In addition, SAVIEE and the Berlin Emotion Database, contain voice clips labeled as Neutral. These datasets also provide clips in different languages and from diverse actors. Further, they are widely used by state-of-the-art methods, thus provide solid baseline evaluations.
6.2 Comparison of Alternate Methods

To validate SCB we utilized a two fold comparison approach. First, we compare against common baseline methods deployed within the SCB framework as replacements for the SCB method (Algorithm 1). For this task, we compare against common classification algorithms such as kNN, SVM and randomForest, as well as boosting and ensemble methods (summarized in Table 2).

Second, we compare against state-of-the-art methods that classify emotion from the same benchmark datasets, namely, Hidden Markov Model [31], Gaussian Mixture Model [37], Orthogonal and Mahalanobis Distance Based [17], Random Deep Belief Networks [35], Simple SVM [29], Adaptive Boosting [28], Softmax Regression [25], CNN [34,3,12], and RNN [13]. These methods are briefly described in Section 2. Results of comparisons with state-of-the-art methods are summarized in Table 3.

6.3 Evaluation Settings

To directly compare to state-of-the-art methods, we utilize parent clip classification accuracy defined as the ratio of True Clip Classifications (TC) and the Total Number of classifications, which includes False Classifications (FC), $\frac{TC}{TC + FC}$.

For consistency, all models are trained and validated using a n-fold cross-validation approach, where $n = 5$. It is important to note that when $\Omega > 0$ adjacent sub-clips overlap, thus share common information. If not controlled, overlap will cause partial over-fitting and artificially higher model accuracy. Therefore, we operate under the constrain that it is critical that sub-clips of the same clip are kept in one fold during model training and testing procedures, thus avoiding any over-fitting.

6.4 Evaluating “Oracle” Implementations

Different “Oracle” implementations discussed in Section 5.3 are evaluated for their relative classification accuracy. The results are illustrated in Figure 4. Overall, classification based on the sub-clip Maximum Emotion Strength yields the best and most consistent results. This finding supports our hypothesis that emotion cues are localized. In addition, it was also observed that SVM was the best performing sub-classification algorithm. Results are further improved after hyper parameter tuning in Section 6.5.

6.5 Hyper Parameter Tuning

6.5.1 Sub-clip Overlap. In the examples in Figures 5a through 5e overlap $\Omega$ displays optimal ranges that present tuning opportunities. Further, the optimal range for $\Omega$, interacts with $N$, as seen for $N = 3$ (Figure 5a), $N = 5$ (Figure 5b), and $N = 7$ (Figure 5c EmoDb). These finding are consistent across datasets as also seen for the Berlin EmoDB for $N = 3$ (Figure 5d), and $N = 5$ (Figure 5c).
Fig. 4: Comparing “Oracle” implementations, using different sub-classification algorithms. Maximum Emotion Strength “Oracle” consistently beats other implementations. SVM is the overall winner as the choice of sub-classification algorithm.

Fig. 5: Classification accuracy when varying Overlap $\Omega$ and number of clips $N$. The charts indicate the existence of optimal values for $\Omega$ and $N$ across datasets.
6.5.2 Number of Sub-clips. One example of tuning for optimal number of sub-clips, is given in Figure 5f. In this example, parent clip classification accuracy peaks between $5 \leq N \leq 7$, while $\Omega = 0.4$. This is consistent with the intuition that more sub-clips increase classification opportunities for the parent clip, but decrease the amount of information in each sub-clip, therefore a trade-off is expected. In addition, the amount of information in each clip also depends on the amount of overlap $\Omega$, the optimization of which is explored in Section 6.5.1.

<table>
<thead>
<tr>
<th>Meta Model</th>
<th>Dataset</th>
<th>kNN</th>
<th>RF</th>
<th>SVM</th>
<th>Naive Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>Berlin EmoDb</td>
<td>0.781</td>
<td>0.789</td>
<td>0.873</td>
<td>0.776</td>
</tr>
<tr>
<td></td>
<td>SAVEE</td>
<td>0.675</td>
<td>0.688</td>
<td>0.735</td>
<td>0.619</td>
</tr>
<tr>
<td></td>
<td>RML</td>
<td>0.666</td>
<td>0.714</td>
<td>0.761</td>
<td>0.645</td>
</tr>
<tr>
<td>SCB</td>
<td>Berlin EmoDb</td>
<td>0.859</td>
<td>0.857</td>
<td>0.880</td>
<td>0.833</td>
</tr>
<tr>
<td></td>
<td>SAVEE</td>
<td>0.739</td>
<td>0.760</td>
<td>0.789</td>
<td>0.714</td>
</tr>
<tr>
<td></td>
<td>RML</td>
<td>0.839</td>
<td>0.825</td>
<td>0.843</td>
<td>0.797</td>
</tr>
<tr>
<td>AdaBoost</td>
<td>Berlin EmoDb</td>
<td>0.735</td>
<td>0.815</td>
<td>0.867</td>
<td>0.791</td>
</tr>
<tr>
<td></td>
<td>SAVEE</td>
<td>0.638</td>
<td>0.700</td>
<td>0.727</td>
<td>0.663</td>
</tr>
<tr>
<td></td>
<td>RML</td>
<td>0.638</td>
<td>0.710</td>
<td>0.726</td>
<td>0.645</td>
</tr>
<tr>
<td>Stacking</td>
<td>Berlin EmoDb</td>
<td>0.787</td>
<td>0.821</td>
<td>0.813</td>
<td>0.787</td>
</tr>
<tr>
<td>Baseline=MetaModel</td>
<td>SAVEE</td>
<td>0.669</td>
<td>0.700</td>
<td>0.723</td>
<td>0.688</td>
</tr>
<tr>
<td>MetaModel(Stack: kNN, SVM)</td>
<td>RML</td>
<td>0.696</td>
<td>0.712</td>
<td>0.731</td>
<td>0.690</td>
</tr>
</tbody>
</table>

Table 2: Comparison of SCB to baseline methods. When used as baselines, classification algorithms (kNN, RF, SVM and Naive Bayes) directly classify the parent clip using its feature vector. In the context of AdaBoost, multiple models using the same algorithm are created using feature subsets to boost the model accuracy. In Stacking, multiple models from different algorithms are combined. In the context of SCB, classification algorithms are used to generate sub-classification (SC) labels.

6.6 Overall Results

In the context of emotion detection SCB is superior to Baseline (Table 2) and state-of-the-art algorithms (Table 3) for all three of the evaluated datasets. We also note that some of the state-of-the-art methods are not directly comparable. In the case of [17], the method eliminates 5% of the data that is labeled as outliers. Similarly, the accuracy for [24] was adjusted based on the number of instances per class. Coincidentally, in the RML EmoDB dataset Happiness had the highest classification error rate, but a lower number of instances. In that case, applying this [25] approach would artificially increase accuracy, however when applied to the SAVEE dataset which has a balanced number of instances, the same method failed to produce similar accuracy improvements. Therefore, in
comparable settings SBC consistently outperforms all state-of-the-art algorithms and baselines.

7 Conclusion and future work

In this study we introduced Sub-clip Classification Boosting (SCB), a novel boosting method based on sub-clip classification. Further we discovered interesting trade-off patterns related to the number of sub-clips and sub-clip overlap. We also introduced and evaluated, several “Oracle” implementations to determine parent clip classifications from its sub-clips. Lastly, we evaluated our SCB framework using 3 benchmark datasets and achieved significant parent clip classification accuracy improvements over state-of-the-art algorithms, as well as consistent generalization across datasets.

<table>
<thead>
<tr>
<th>Method</th>
<th>Berlin</th>
<th>RML</th>
<th>SAVEE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCB</td>
<td>0.880</td>
<td>0.843</td>
<td>0.789</td>
</tr>
<tr>
<td>Hidden Markov Model [31]</td>
<td>0.830</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gaussian Mixture Model [37]</td>
<td>0.81</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Orthogonal and Mahalanobis Distance Based</td>
<td>0.820*</td>
<td>0.810*</td>
<td>0.830*</td>
</tr>
<tr>
<td>Random Deep Belief Networks [35]</td>
<td>0.823</td>
<td>0.734</td>
<td></td>
</tr>
<tr>
<td>Simple SVM [29]</td>
<td>0.563</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adaptive Boosting [28]</td>
<td>0.708</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Softmax Regression [25]</td>
<td>0.790 (0.879**)</td>
<td>0.734</td>
<td></td>
</tr>
<tr>
<td>CNN [34,3,12]</td>
<td>0.845***</td>
<td>0.69[12]</td>
<td></td>
</tr>
<tr>
<td>RNN [13]</td>
<td>0.695</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Accuracy Evaluation of the SCB Framework. *Method eliminates 5% of the data. Accuracy not directly comparable. **Accuracy was adjusted based on number of instances per class, and controlling for speaker, therefore not comparable. *** Across speakers accuracy varied from 0.84 to 0.845 for the Berlin EMODb. One study [12] used the SAVEE dataset in addition to Berlin EMODb.

References