Emotions recognition using binary fingerprints

Xavier Anguera, Esperança Movellan* and Miquel Ferrarons*

Telefonica Research,
Edificio Telefonica-Diagonal 00, 08019, Barcelona, Spain,
xanguera@tid.es

Abstract. Recognizing emotions in speech recordings is a topic rapidly gaining popularity in the scientific community which has interesting challenges in terms of research and potential applications in a real world scenario. In this paper we focus on the identification of the predominant emotion in a speech excerpt among a set of possible (previously known) emotions. In most prior art research this problem is approached by using acoustic features directly derived from the audio (sometimes expanded to several thousand components) and using standard machine learning techniques to construct emotion models for each of the known emotions. In this paper we propose the use of an alternative approach recently introduced for speaker modeling which brings the feature space, modeling and classification to the binary space. Experiments show that the proposed method obtains very competitive results using standard (low dimensional) features.

Keywords: Emotions recognition, binary fingerprints, classification, modeling

1 Introduction

Recognition of the emotions reflected in an audio signal is a quite recent field of research that is quickly gaining popularity. Indeed, this topic is not only interesting from the point of view of research, with several novel unanswered challenges, but also in a real world scenario, with multiple applications in call-centers or monitoring user interaction with automated dialog systems.

Most research performed until now in this area [5, 6] has been addressed as a classification problem where an input utterance is classified among a set of two (neutral vs emotional) or several known emotions (e.g. anger, sadness, happiness, etc.). Statistical machine learning techniques used to model each of the emotions range from Support Vector Machines (SVM) [7], Neural Networks (NN) [3], Gaussian Mixture Models (GMM) [1], Hidden Markov Models (HMM) [2], among others. Currently, in order to obtain state-of-the-art results usually

*Esperança Movellan and Miquel Ferrarons were visiting students from Universitat Pompeu Fabra at the time of this work.

hundreds (or even thousands) of features are stacked together into feature vectors that capture the dynamics and acoustic characteristics of each emotion. Many times a single feature vector ends up being used for an input utterance as a result of the application of various statistics on short-term features. Such high number of features is usually not easy to interpret and it is not ensured whether all information contained in the short-term feature vectors is retained or not. In order to reduce features dimensionality some works like [4] perform feature selection to select those features that are most prominent in differentiating among emotions, which still does not answer concerns about interpretability and retainment of information.

In this paper we propose a departure from common practices in two ways. On the one hand, we use a totally different modeling approach by constructing binary fingerprints from each emotion and then comparing them entirely in the binary domain. This modeling approach was initially been proposed in [10] as a way to model speakers by their voices and is the first time it has been successfully applied to emotion recognition. On the other hand, our system uses standard acoustic features (MFCC 39-dimensional at this point, but this is not a limitation of the system) which are not compressed into a single feature vector at feature level. Even though such features are of much lower dimensionality and only represent a very local portion of the signal, with binarization we are able to indirectly incorporate long term information and successfully model emotions. Furthermore, given that the emotion fingerprints are represented by a binary vector it is much simpler to interpret how an emotion is represented and see the differences between different emotions.

Experimental results show that our approach can achieve very competitive results in comparison to results obtained with a well known emotion recognition system available online, while using much simpler features.

2 Binary Emotion Classification Algorithm

The proposed emotion classification algorithm is based on the binary speaker modeling algorithm initially proposed in [10]. It is comprised of three main modules as seen in figure 1. First, an offline module trains a KBM (binary Key Background Model) on as much acoustic data as possible. The KBM model is a GMM model whose function is to perform the change of base between the input acoustic features into binary form. Second, emotion models are trained given the data available for each emotion. To do so, the data is first converted into binary form using the KBM model and then a binary fingerprint is obtained. The resulting binary fingerprint models are small to store and, as will be seen later, fast to compare. Third, an online test module uses the KBM model to transform any input utterance into binary domain and then compares them to the emotion fingerprint models, selecting the one with highest similarity. Further details in each of these modules is given next.

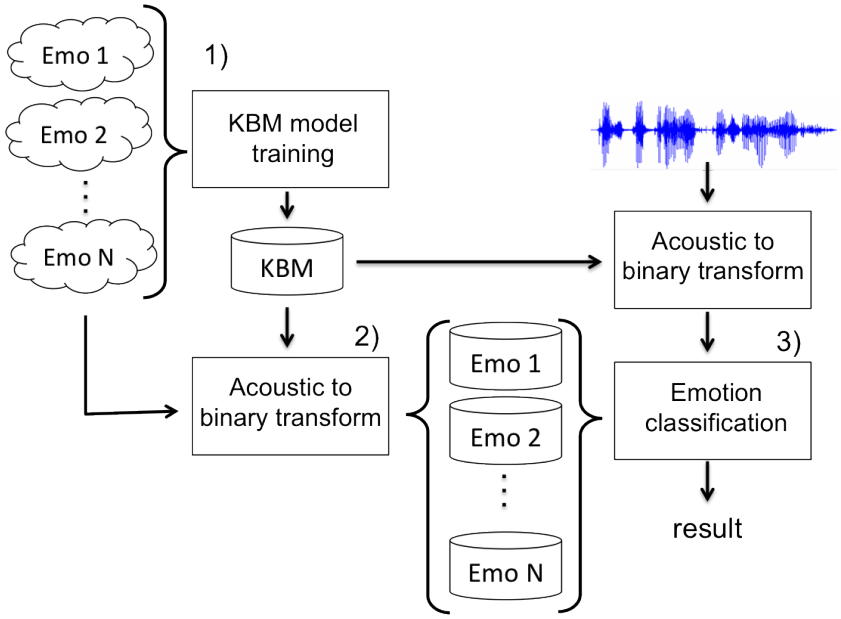

Fig. 1. *Binary emotion classification algorithm, comprised of three main modules: 1) KBM training; 2) emotion fingerprint training; 3) classification of input utterances.*

2.1 KBM Training

The KBM model is used to transform acoustic feature vectors into binary form. The KBM model differs from a standard GMM model in the objective the model is created for. While in a generative model when training one wants to optimize the representation of the data given the model, the objective of the KBM model is to optimally segment the acoustic space so that small differences in the input feature vector derive in very different binary fingerprints only if these differences are important. Many alternatives can be used to train the KBM model (for example [11]). In this paper we use a simple EM-ML training using Gaussian initialization through Gaussian splitting. In order to train the KBM appropriately we need training data that covers all emotions we want to recognize, all within similar acoustic conditions that the ones we test on. In this paper we do not use any prior information regarding which emotion each training file belongs to and use all acoustic data in a single training pool. For this reason we considered using the same database used in test, as there is not danger of data overfitting.

Once we have the trained KBM model, for each acoustic feature vector (usually extracted every 10ms from the audio signal) we use the model to obtain a binary feature vector whose dimensionality corresponds to the number of Gaussians used in the KBM model, existing a direct relation between each Gaussian in the model and a position in the feature vector. We set to 1 the positions in the vector corresponding to the closest Gaussians to that feature vector, where

distance to a Gaussian is measured in terms of likelihood of feature vector x being modeled by that Gaussian λ, i.e. $Lkld(x|\lambda)$. Intuitively, by doing this binarization we are reducing the information we keep from the feature vector to encode only the region in the acoustic space where such feature occurs. Such regions come from quantizing the acoustic space using the KBM model so that when we encounter a high density of feature vectors in training data we quantize the space more densely, and vice-versa. After some experimentation we got to the conclusion that regardless of any other parameters or application, selecting around 5 closest Gaussians per feature vector always achieved always good performance. This base transformation results in a very sparse binary vector that can be easily compressed if need be for storage or transmission.

2.2 Emotion Binary Fingerprints Computation

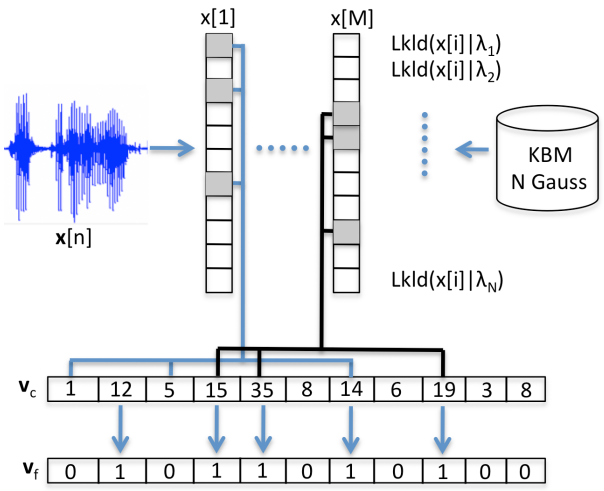

Fig. 2. *Fingerprint training algorithm.*

Figure 2 shows the overall process used to compute a binary fingerprint for an emotion given some acoustic training data. This is the same process followed to convert an input utterance into a single vector binary form for classification into an emotion. In fact, the block in Figure 1 titled *acoustic to binary transformation* corresponds to the algorithm is Fig. 2 preceded by an acoustic feature extraction step to convert the input audio signal into the desired intermediate acoustic feature vectors.

The different steps involved in the algorithm shown in Fig. 2 are explained next. First, given a set of training utterances summing M acoustic feature vectors $x[i]|i = 1 \ldots M$ we convert them into binary form using the KBM model as explained in the previous section. We then count how many 1 values we obtained

for each position in the M binary vectors and save such count into the v_c vector, which has the same dimension as the binary feature vectors (i.e. the number of Gaussians in our KBM model). The v_c vector contains the counts histogram indicating which Gaussians are closest to the M given input feature vectors. The last step in the process is to convert these counts vector into a binary form. In this case we choose the 25%-highest count dimensions to become 1, setting to 0 the rest. The resulting binary fingerprint has the size of the number of Gaussians in the KBM and is usually several orders of magnitude smaller than any traditional GMM or SVM model.

Note that what this step actually does is an averaging of the information obtained at feature level regarding where in the acoustic space the input features mostly occur. This is very similar to the averaging performed by state-of-the-art emotion recognition systems at feature level by representing the whole utterance using several functionals computed from the set of acoustic features extracted from the signal, like in our case, at short-term intervals. The main difference here is that as each bit in the fingerprint directly represents the activation status of a given KBM Gaussian, and each Gaussian represents a regions in the multidimensional space where the input acoustic feature vectors reside, we can directly obtain a relationship of which acoustic regions are most prevalent for each one of the emotions, being able to more easily interpret the results.

2.3 Emotions Identification

Given an input signal we want to classify into the emotion its most closely reflects, we first convert such signal into binary form and then compare it with the set of finite emotions for which we have binary fingerprints. As mentioned before, the binarization of the test utterance is done in the same way as with the training of emotion fingerprints, i.e. first obtaining an intermediate acoustic feature vector representation (the same in which the KBM model was trained) and then mapping these features into binary space and summarizing the test utterance using a single binary vector.

The comparison of the input binary vector with the available emotions is done in the binary space by using the similarity measure shown in 1. Similarities obtained range from 0 to 1, the higher the closest. Note that given that all the processing after binarizing the feature vectors is performed on the binary space, it is very fast to compute.

$$S(\mathbf{v}_{f1}, \mathbf{v}_{f2}) = \frac{\sum_{i=1}^{N}(v_{f1}[i] \wedge v_{f2}[i])}{\sum_{i=1}^{N}(v_{f1}[i] \vee v_{f2}[i])} \tag{1}$$

3 Experiments

In this section we describe the experimental setup and the results of our proposed system, compared to state-of-the-art alternatives.

3.1 Experimental Setup

In order to test the proposed algorithm we used the publicly available emoDB database [8]. The database is freely available online [1] and consists of 535 short recordings where seven emotions (Anger, Boredom, Disgust, Fear, Happiness, Sadness, and Neutral) are simulated by several German actors. Given the small size of the database most researchers (including us) perform a 10-fold cross-validation experiment, where 9/10 of the database is used for training models and the rest is used for testing, which results in 10 tests whose results are averaged to obtain a single overall result. For each partition we use an equal number of files from each emotion. Note that the exact way in which partitions should be constructed is not commonly defined, therefore it is difficult to compare results among research papers as each particular distribution might give very different results. Such results are evaluated both using a non-weighted accuracy metric (i.e. the percentage of files for which the emotion has been correctly classified) and a weighted accuracy metric where each individual emotion accuracy is inversely weighted by its a priori probability within the database. When using emoDB the weighted metric is necessary as a different number of files is available for training each emotion.

To validate our proposed system we compared the results with the open-source OpenEar package [7], which has been proven to obtain state-of-the-art results using SVN-based models for each emotion. The software was downloaded from [2] and default parameters were used to perform model training and recognition, with the exception of the front-end configurations that were not available in the original package, which were computed by modifying the example configuration scripts.

The selection of the acoustic features used for the experiments needs some explanation as it differs slightly between our proposed system and the openEar package. The openEar package offers several configuration files that extract a set of acoustic feature vectors ranging from PLP, MFCC, Pitch and others at fixed short-term intervals on the signal and later computes a set of functionals to obtain a single feature vector for the whole input utterance (both for training and testing utterances). The dimensionality of the resulting vectors varies according to which exact functionals are used (e.g. Extremes, Regression, Moments, Percentiles, Crossings, Peaks, Means, etc.). For these experiments we used the parameter sets labelled as *emo_base (*988 *dimensions), emo_large (*6552 *dimensions) and emo_IS09 (*384 *dimensions, used in the 2009 emotion recognition challenge [9]). In addition, we created* 2 *extra set of features to better compare results with our proposed algorithm. The first feature set corresponds to a 39-dimensional MFCC vector extracted every* 10*ms and collapsed into a single vector per utterance using the mean functional. The second feature vector corresponds to the same initial features (39-dimensional MFCC) to which we applied the same functionals used in the emo_IS09, obtaining a single vector os* 469 *dimensions.*

[1] http://http://pascal.kgw.tu-berlin.de/emodb
[2] http://sourceforge.net/projects/openart

For our system we have chosen to initially test the system using a standard feature vector used in many areas of speech processing. The feature vector corresponds to MFCC-39 which includes 12 MFCC features + energy, their deltas and double deltas. These were obtained from the signal by using a 25ms Hamming-windowed signal every 10ms. Note that these features are extracted with a different front-end than the one than comes with openEar, although all parameters have been set equally with the exception that in here we initially do not perform a mean of the input features into a single vector. In addition, in order to better compare our system with openEar, we extracted features using the openEar front-end and using the appropriate configurations to obtain the emo_base, emo_large and emo_IS09 sets, but without applying any functional and using all short-term feature vectors as input to our system. These result in feature vectors of dimension 53, 171 *and* 35 *respectively.*

3.2 Results

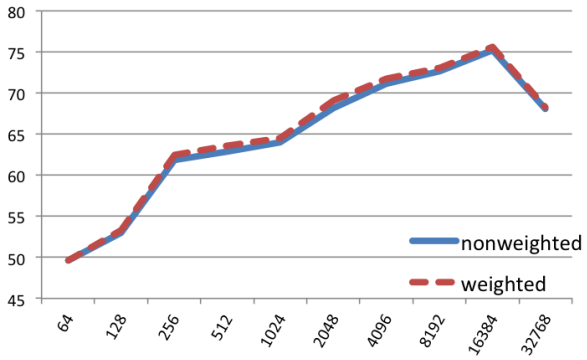

Fig. 3. *Binary fingerprint accuracy plot for different number of bits per fingerprint.*

In an initial test we evaluate the performance of our binary fingerprinting system with respect to the dimensionality of the binary vector, i.e. the size of the KBM model. To do so, we train a KBM using all data available from all partitions of the emoDB database. Then, for each partition we obtain the emotion fingerprint for all 7 *emotions and test them on the test data for that partition. Results are shown in Fig. 3 for multiple of* 2 *KBM sizes. We observe how the weighted and non-weighted accuracy metrics are both giving very similar (almost identical) results. For both cases, accuracy keeps increasing until* 2^{14}*. We hypothesize that beyond that point there might be not enough data in the training database to correctly and consistently partition the acoustic space. In fact, when performing the EM-ML iterations we observe very small Gaussian variances for many of the Gaussians, which makes them very unstable when performing the conversion to binary form. Although optimum results are obtained well beyond*

2048 *dimensions, we use this dimensionality in the comparisons from this point on as it seems to be a good tradeoff between complexity and accuracy.*

Table 1. *Comparison of results with 10-fold cross-validation testing*

system	features	weighted acc.	non-weighted acc.
OpenEar	MFCC39	56.27%	57.58%
	MFCC39 + IS_09 stats	66.52%	65.4%
	IS_09	67.77%	65.79%
	emo_base	72.45%	70.84%
	emo_large	72.24%	70.86%
Binary	MFCC39 (2048 dimensions)	68.20%	69.14%
	MFCC39 (16384 dimensions)	**75.23%**	**75.62%**
	IS_09 (2048 dimensions)	59.64%	61.63%
	emo_base (2048 dimensions)	65.20%	65.91%
	emo_large (2048 dimensions)	67.98%	66.99%

Next, Table 1, shows the comparison between openEar and our system on the different feature vectors described above. On the upper part of Table 1 we show results for openEar. As expected, results on the MFCC39 feature vector are quite poor. This might be due to the limited number of dimensions available for the SVM-based modeling and the loss of information derived from performing only an average functional on the data. When adding extremes, regression coefficients and moments (MFCC39 + the IS2009 functionals) results get much improved. Furthermore, if we use the predefined features provided by the openEar package we get the highest results, being the emo_base features the best performing with the smallest dimensionality. Note that the results for emoDB that we are obtaining are considerably lower than those advertised in papers like [7]. While we believe we are using the same setup as they do in their experiments (with the exception of the lists used for each partition) to date we have still not been able to replicate their results.

On the lower part or Table 1 we show results using our proposed binary fingerprint system on the same feature vectors. Unlike with the openEar system we are able to extract much more information out of the MFCC39 features, which obtain very good results. From all values in Figure 3 we have chosen the ones for 2048 *and* 16384 *dimensions, both obtaining results exceeding those shown for openEar. Opposite from openEar, when using any of the feature vectors that are commonly used in emotion recognition we are not able to achieve as good results as with MFCC39. We hypothesize that the reason for this is that with a reasonable binary dimensionality (*2048 *is used in these tests) it might not be enough for the KBM to properly quantize the acoustic space of such high-dimensional acoustic feature vectors, even if these might not include much usable information in some of the dimensions. It remains as future work how to be able to successfully process these high-dimensional features to take full advantage of all the information they contain.*

Over all, we can see how using simple MFCC39 features we achieve similar (or better) results that using the standard high-dimensional feature vectors currently used in emotion recognition.

4 Conclusions and Future Work

Recognizing the most prominent emotion that arises from a given acoustic utterance is a task which has recently attracted the interest of the research community and has clear implications and application in the commercial world. Typically, emotion recognizers are classifiers that select which emotion model better fits the acoustic data. Models are trained using standard machine learning approaches and feature vectors are usually obtained by stacking together several acoustically-derived parameters together, and postprocessed by functionals that model the evolution of these parameters over time. In this paper we propose a novel approach for emotion classification that has been recently introduced for speaker modeling. Its main characteristic is that emotions modeling and search for the best matching emotion is all done in the binary domain. This is the first time we demonstrate that this novel method can be suitable beyond speaker modeling and show that with simple input feature vectors we can achieve results that are in many occasions better than those of state-of-the-art emotion recognition systems.

Future work will involved testing the system with different acoustic features to find an optimal combination of features to give optimum results, and the test of the system on bigger and more challenging databases.

References

1. *I. Luengo, E. Navas and I. Hernaez: Combining spectral and prosodic information for emotion recognition in the Interspeech 2009 Emotion Challenge. In Proc. Interspeech 2009, Brighton, UK, 2009.*
2. *R. Huang and C. Ma: Towards a speaker-independent Real-Time Affect Detection System. In Proc. International Conference on Pattern Recognition, 2006, pp.1204-1207.*
3. *M.W. Bhatti, W. Yongjin and G. Ling: A neural network approach for human emotion recognition in speech. In Proc. International symposium on Circuits and Systems, 2004, pp II-181-4 Vol. 2.*
4. *Jia Rong, Gang Li and Yi-Ping Phoebe Chen: Acoustic feature selection for automatic emotion recognition from speech. In Information Processing and Management Vol. 45, 2009, pp. 315328.*
5. *S. G. Koolagudi and K. S. Rao: Emotion recognition from speech: a review. In International Journal of Speech Technology, no. July, Jan. 2012.*
6. *M. El Ayadi, M. S. Kamel, and F. Karray: Survey on speech emotion recognition: Features, classification schemes, and databases. In Proc. International Conference on Pattern Recognition, vol. 44, no. 3, pp. 572-587, Mar. 2011.*
7. *Florian Eyben, Martin Wllmer, and Bjrn Schuller: openEAR - Introducing the Munich Open-Source Emotion and Affect Recognition Toolkit. In Proc. Interspeech 2009, Brighton, UK.*

8. *F. Burkhardt, A. Paeschke, M. Rolfes, W. Sendlmeier, and B. Weiss: A database of german emotional speech. In Proc. Interspeech 2005, Lisbon, Portugal, pages 1517 1520, 2005.*
9. *Bjorn Schuller, Stefan Steidl and Anton Batliner: The INTERSPEECH 2009 emotion challenge. In Proc. Interspeech 2009, Brighton, UK, 2009.*
10. *Xavier Anguera and Jean-François Bonastre: A novel speaker binary key derived from anchor models. In Proc. Interspeech 2010, Makuhari, Japan, 2010.*
11. *Xavier Anguera: Speaker Independent discriminant feature extraction for acoustic pattern-matching. In Proc. ICASSP 2012, Kyoto, Japan, 2012.*