Automatic speech feature learning for continuous prediction of customer satisfaction in contact center phone calls

Carlos Segura[1], Daniel Balcells[12], Martí Umbert[13], Javier Arias[1], and Jordi Luque[1]

[1] Telefonica Research Edificio Telefonica-Diagonal 00, Barcelona, Spain
[2] Dept. Signal Theory and Communications,
Universitat Politècnica de Catalunya, Barcelona, Spain
[3] Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain
carlos.seguraperales@telefonica.com
jordi.luqueserrano@telefonica.com

Abstract. Speech related processing tasks have been commonly tackled using engineered features, also known as hand-crafted descriptors. These features have usually been optimized along years by the research community that constantly seeks for the most meaningful, robust, and compact audio representations for the specific domain or task. In the last years, a great interest has arisen to develop architectures that are able to learn by themselves such features, thus by-passing the required engineering effort. In this work we explore the possibility to use Convolutional Neural Networks (CNN) directly on raw audio signals to automatically learn meaningful features. Additionally, we study how well do the learned features generalize for a different task. First, a CNN-based continuous conflict detector is trained on audios extracted from televised political debates in French. Then, while keeping previous learned features, we adapt the last layers of the network for targeting another concept by using completely unrelated data. Concretely, we predict self-reported customer satisfaction from call center conversations in Spanish. Reported results show that our proposed approach, using raw audio, obtains similar results than those of a CNN using classical Mel-scale filter banks. In addition, the learning transfer from the conflict detection task into satisfaction prediction shows a successful generalization of the learned features by the deep architecture.

Index Terms: feature learning, end-to-end learning, convolutional neural networks, conflict speech retrieval, automatic tagging.

1 Introduction

Nowadays, call centers are one of the most used customer interaction channels. Beyond the actual content of the call, nonverbal communication in speech can be perceived and interpreted from a social and psychological point of view. Such social constructions influence and shape our perception for contentedness and perceived levels of engagement and cooperation over the contact interaction. Automatic understanding and retrieval of

a person's social phenomena from speech is of special interest in multiparty conversations. Conflict is recognized as one of the dimensions along which an interaction is perceived and assessed. Having appropriate technology for speech mining and information retrieval is a prior step for understanding customers needs, their expectations and thus for the improvement of the service. Therefore, automatic estimation of customer satisfaction based on the audio of a phone contact is clearly of interest for any business.

Most of the approaches for task classification involving audio signals usually rely on traditional features, such as spectral based ones, and use a decoupled approach which comprises mainly two steps. In the first step, features are extracted from the raw audio signal, e.g., front-end processing, and then employed as an input for a second step that carries out the model learning. Usually, such features are designed by hand and strongly depend on a high expertise both on the knowledge of the addressed problem and on the audio signal itself.

In recent years, a trend has arisen in different research fields that aims at building architectures capable of learning features directly from the raw input data. For instance, in computer vision novel feature learning techniques are applied directly on the raw pixel representations of images avoiding the signal parameterization or any other prior preprocessing [15]. Budnik *et al.* [3] report an extensive comparison of the performance of CNN based features with traditional engineered ones, as well as with combinations of them, in the framework of the TRECVid semantic indexing task.

In speech processing, several successful DNN-based systems for extracting frame-by-frame speech features have been presented [9,4]. Novel proposed features, usually extracted from DNN topologies, are reaching the same level of proficiency and success as those from the image or video processing fields. Nevertheless, there are few works that address the processing in one single step and most of them rely on classical spectral representations of the speech signal from which they compute DNN-based features. Some recent works aim to directly handle the raw audio data in one single step. Jaitly *et al.* [12] modeled speech sound waves using a Restricted Boltzmann machine (RBM) and reported better phoneme recognition performance than methods based on Mel cepstrum coefficients. In [1,18] the authors show that CNNs can model phone classes from the raw acoustic speech signal, reaching performance on par with other existing feature-based approaches. Previous work in [6,10] has inspired our approach for end-to-end learning of salient features suitable for conflict detection in dyadic conversations. Hoshen *et al.* [10] trained a system end-to-end rather than manually designing the robust feature extraction for the multichannel input, jointly optimizing a noise-robust front-end along with the context-dependent phone state DNN classifier.

Additional work has applied computational analysis to recorded call center conversations. Park and Gates [19] and Zweig *et al.* [24] proposed machine learning techniques using linguistics and prosodic features to predict customer satisfaction. Emotion detection has also captured the attention of researchers within the call center domain. We can find other examples given by Devillers *et al.* in [5] where they report results on detecting emotional states by using acoustic features; or in [22] detecting negative emotions in call center conversations and its suitability for inferring customer satisfaction.

In this paper we propose a feature learning approach based on Deep Convolutional Neural Networks (DCNN) for continuous prediction of satisfaction in contact center

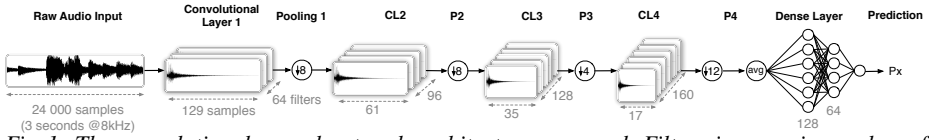

Fig. 1: The convolutional neural network architecture proposed. Filter sizes are in number of samples, and number of units are depicted in the scheme. Method and value for pooling layers using, decimation through max-pooling (↓) and averaging (avg) which compute some stats as the mean and standard deviation, are also reported. Training samples are randomly selected from 3 seconds excerpts from original raw data.

phone calls. The network is initially trained on debates from French TV shows [21] aiming to find out salient information in raw speech that correlates with conflict level over the conversation. Then feature transfer is assessed on completely unrelated data, that is, phone calls from several contact centers. It is done by adapting the parameters of the ending layers with few samples in order to simulate a real low-resourced scenario. We analyse more than 18 thousand phone conversations made to the call center of a major company in a Latin American country. All the calls were made in Spanish. The proposed system should not only specifically tackle the conditions present in telephone channel, but also be able to discover suitable and robust descriptors from the raw audio data which at the same time are meaningful and both domain and task agnostic.

2 End-to-end learning

2.1 System Description

The CNN architecture that we used as a basis for all our experiments is depicted in Figure 1. CNN have reported great success achieving traslation invariance in image recognition tasks [16,14]. In the same sense, our deep CNN-based feature learning architecture makes use of local filtering and feature pooling. It comprises a total of 6 layers: four convolutional layers with different amounts of filters and lengths (see Figure 1 for layer sizing) alternating with decimation and max pooling layers. The decimation value is 8 for the first two convolutional layers and 4 for the third one. Average pooling is used at the output of the fourth convolutional layer.

Let's assume the speech input to the CNN is a matrix, is $\boldsymbol{x}$, whose columns are raw audio vectors $\boldsymbol{x} = [\boldsymbol{x_n}, \boldsymbol{x_{n+1}} \dots \boldsymbol{x_{n+N}}]$ where $\boldsymbol{x}_n$ is the audio sample vector shifted by a stride. In this work we used a value of 1 for time shifting and the $\boldsymbol{x_n}$ vectors range in size from 3 seconds to several minutes at $8,000$ samples per second. The activations at the first convolutional layer comprise $J = 64$ filters and we denoted them as $\boldsymbol{h}_j = [h_1 \quad h_2 \cdots h_J]$. Therefore, the convolutional layer operation can be seen as a convolutional operation of each filter on the input raw audio,

$$h_j = \theta\left(\boldsymbol{w}_j \boldsymbol{x}^T + b_j\right),$$

where $\theta(x)$ is the activation function and b_j the bias term for filter h_j. Note that it can be seen as a standard MLP network layer taking into account that filters are of

the same size and input raw audio shifted accordingly. Successive convolutional filters are applied on two dimensions rather than only one as in the input layer. In addition to convolutional filters, max-pooling layers perform local temporal max operations over the input sequence, selecting the maximum in a window of decimation d size, as ($\downarrow d$) in Figure 1. More formally, the transformation at starting sample vector n, $\boldsymbol{c}_n^j$, corresponding to the filter output sequence of the first convolutional layer and jth filter is:

$$\max_{n-\frac{(d-1)}{2} \leq s \leq n+\frac{(d-1)}{2}} \boldsymbol{c}_s^j$$

Next, the average pooling output, see Figure 1 at the CL4 output feeds two dense layers. Latter pooling stage compacts even more the original signal by computing some stats such as maximum, mean and variance from the CNN architecture output. The dense layer is employed as a back-end for the modeling of the salient features computed by previous convolutional steps. It comprises an input layer of 128 and an output one with 64 neuron units, respectively. In the first two convolutional layers no activation function θ is used, the subsampling/stride of the convolution being the only non-linearity. At the third and fourth layers, rectified linear units along with max-pooling are applied. Finally, both dense layers use maxout [8] as activation function and a dropout value of 0.5.

The network is trained using stochastic gradient descent (SGD) with mini batches of size 200. In the case of regression task, that is, for the SC^2 conflict corpus, we optimize network parameters based on the Mean Square Error (MSE) computed between network outputs and labels. We have experimented with different window sizes, but mainly we select t seconds excerpts randomly picked from original raw audio signal. For example, 3 seconds excerpts lead to sample instances that amount for 24 thousand raw audio samples at 8kHz sampling rate. Note that conflict scores are given for each 30s excerpt within the SC^2 corpus (see section 3.1) and thus our sampling procedure may extract training instances that are non-relevant regions within the 30s excerpts, like silences or other non-speech related content such as background music. Nevertheless, we assume that the effect of training the network by using some of those 3 second "noisy" instances is mitigated by the mini batch size, the slice context and the number of epochs employed for the network training. Note that in the case of call center phone calls, audio recordings are stereo and at 8kHz sampling rate, so we decided to downsample original SC^2 corpus to meet telephone channel bandwidth. This aims at mitigating sampling rate effects on the training and feature transferring between both systems. We also applied a simple speech detection algorithm based on energies for discarding likely non-speech samples.

2.2 Filter learning and spectogram based features

Figure 2 represents the normalized magnitude spectra taken from the first convolutional layer CL1 (see the Figure 1). Filters of length 128 samples are sorted by the frequency bin containing the peak response.

It is worth to note that the filters response is almost linear in the range $[0, 1000]$Hz, which concentrates almost 80% of the convolutional filters. From this point on, normalized magnitude behaves more like a logarithm function, until it reaches 4kHz fre-

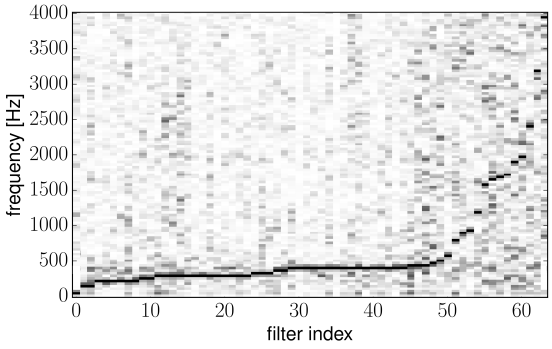

Fig. 2: Normalized magnitude spectra of the filters in the first convolutional layer CL1, see Fig. 1, ordered by frequency bin containing the peak response (black pixels).

quency, after which filter peaks are more widely spaced. Similar filter bank responses are reported in previous works [6,10].

In order to compare previous end-to-end feature learning systems, we consider a second approach, that is, classical spectrogram "images" as the network input. To do so, we try to maintain the same network architecture and parameters, freezing the number of layers and units in each of them and the learned parameters, in order to have fair comparisons between both systems. The input to the network consists of the output of 128 MEL-scale log filter banks energies. Therefore, the convolution is only performed in the time dimension using filters of 5x128 coefficients at the first layer CL1, 5x1, 4x1 and 4x1 in the following CL layers. Latter average-pooling step and dense layers match the configuration used within the raw-audio based system.

2.3 Feature Transfer

Once network parameters are trained through the SSPNet Conflict Corpus (SC^2), we make use of the same network parameters as initialization but only for those in the dense layers. We are interested in adapting the network to a different task, as predicting self-reported customer satisfaction, but also in assessing the suitability of the filter banks learned in the convolutional steps. Therefore, we decide to keep frozen the parameters from all convolutional layers, that are more focus on the signal representation, at the same time we perform stochastic gradient descent (SGD) only on the parameters within the dense layers. The key idea is to avoid retrain the whole system from scratch, incorporating previous learning parameters as initialization values for the new network training and performing SGD with a small learning rate and lesser number of epochs. Thus speeding up the network training since we do not need to discover again a new feature representation of the audio signal.

3 Experimental Evaluation

Two different data sets have been employed in the following experiments. The former comprises a collection of political debates, from TV shows in French, manually labelled from several annotators in terms of conflict scoring. The second is a collection of phone call conversations in Spanish with self-reported customer satisfaction from several contact centers in a Latin American country. Experiments and simulations have been carried out using Theano library [2].

3.1 Corpus

The SSPNet Conflict Corpus (SC^2) was collected in the framework of the Social Signal Processing Network (SSPNet), the European Network of Excellence on modelling, analysis and synthesis of non-verbal communication in social interactions [23]. SC^2 corpus was used in the conflict challenge organized in the frame of the Interspeech 2013 computational paralinguistic challenge [13]. It contains $1,430$ clips of 30 seconds extracted from a collection of 45 Swiss political debates (in French), 138 subjects in total (23 females and 133 males). The clips have been annotated in terms of conflict level by roughly 550 assessors, recruited via Amazon Mechanical Turk, and assigned a continuous conflict score in the range $[-10, +10]$. The binary classification task is created based on these labels, namely to classify into high (≥ 0) or low (< 0) level of conflict. The metrics used in the Conflict detection challenge were Unweighted Average Recall (UAR) and Correlation Coefficient (CC) that we also report in this work.

The call center data is composed of $19,360$ inbound phone calls [17]. It represents a random subset of calls extracted from contact centers in one Latin American country. All the calls were made in Spanish. Data was collected throughout one month such that it comprises a huge variety of interactions between the customer and the call center's agent and different client's requests. At the end of each call, the customer is called back and gently asked to complete a survey related to the service:

According to its previous call to our call center, how satisfied, overall, are you with the telephone service of X. Press 1-5 where 1 is very dissatisfied and 5 very satisfied.

The table 1 reports the distribution of customer satisfaction in our data set. Note that the distribution is significantly skewed towards 5 (64%). Calls with low level of satisfaction, 1 and 2, are approximately 17% of the data. For this data set, a binary classification task is created based on these labels, namely to classify into high satisfaction (≥ 4) or low satisfaction (≤ 2) level of satisfaction.

	Self-reported customer satisfaction				
	1	2	3	4	5
Number of calls	2651	648	1229	2412	12420

Table 1: Distribution of the self-reported satisfaction score in the call center data set.

	CC	UAR
Spectrogram CNN	0.793	0.784
Raw-audio CNN	0.779	0.798
SVR SMO [21]	0.826	0.808
Ensemble SPLS Nyström [11]	0.849	-

Table 2: Results on SSPNet Conflict test database in terms of Unweighted Average Recall (UAR) and Correlation Coefficient (CC).

3.2 SSPNet Conflict Challenge

The baseline system provided by the organizers in the Conflict Sub-challenge, based on support vector regression, reached a remarkable performance around 80.8% UAR on test conditions [21] for the binary classification task. Previous result was overcome by the conflict sub-challenge winner [20], using random K-nearest neighbors and reaching 82.6% of accuracy and further improved at following reported works [11]. Latter paper reported 84.9% accuracy based on an ensemble Nyström method for high-dimensional prediction of conflict. It aimed to model a set of $6,373$ features, comprising a variety of low-level descriptors and functionals, extracted per conversation basis using OpenSmile [7]. For further details, see the table II in [11].

The table 2 reports the results obtained with the proposed systems, raw-audio-based and spectrogram-based CNNs, in terms of the metrics initially proposed in the SSPNet challenge: Unweighted Average Recall (UAR) and Correlation Coefficient (CC). The estimation of the conflict level in each test audio file is performed running the trained network using windows of 3 seconds with an overlap of 50% yielding to a time series of 19 conflict values. Finally, those conflict values are averaged to get the final score. The table shows that both systems perform similarly in the two conflict tasks, regression and classification, and that both are comparable to the baseline in the third row. It is worth to mention that we have not exhaustively tuned our CNN architecture to optimize CC and UAR metrics and that the original signal is downsampled from 48kHz down to 8kHz prior any further processing. The main conclusions from the table are 1) our end-to-end learning scheme is able to discover salient features directly from raw audio, 2) it performs comparable to reported systems based on traditional hand-crafted features and 3) there is no significant difference between using raw audio or Mel-like showing the ability of CNNs to learn from raw audio.

The main difference between the methods is the computational cost of the front-end. In the training phase of the spectrogram-based approach, the features are precomputed once to speed up the process. On the other hand, the raw-audio based front-end has

	Max	Min	Mean	Median
Random Initialization	0.56	0.553	0.585	0.600
Transfer Learning	0.537	0.540	0.567	0.584

Table 3: Comparison of the performance of the continuous satisfaction estimation in terms of AUC metric using random weights initialization and using the weights learned from the SSPNet conflict data.

	30s	60s	120s	240s
Random Initialization	0.552	0.534	0.550	0.600
Transfer Learning	0.520	0.516	0.542	0.584

Table 4: Comparison of AUC scores for the raw-audio CNN systems tested only on the last 30, 60, 120 and 240 seconds of every call.

more computational cost because it must run at each iteration, since it is trained jointly with the classifier in the CNN. However, in the testing phase, we see a speed increase of 1.64x in the case of the raw-audio approach due to the fact that it runs entirely on the GPU, while the spectrogram extraction step is computed in the CPU in our setup.

3.3 Feature Transfer and Satisfaction Prediction

The original SC^2 corpus is downsampled since one of the main purposes of this work is to validate that our learned features generalize to the task of estimating self-reported customer satisfaction, thus ensuring that the learned features can be applied to telephone channel conditions. Moreover, working with signals at 8kHz also makes it easier to train the CNN with raw audio input. The proposed raw-audio CNN system is applied on the task of predicting the customer satisfaction from call center conversations. The main difference with respect to the SSPNet Conflict detection task is that the network will be trained for a binary classification problem using a logistic regression loss function. The training recordings consist of 200 conversations, with a balanced number of low and high satisfaction. Only the second half of the phone call is used for training the CNN. To assess the performance on held-out data, a balanced testing subset of 1000 conversations is prepared. In addition, aiming to study the effects on the duration of the analyzed time, we focus our analysis on the ending part of the phone call conversation.

Table 3 compares the performance in terms of AUC for the two considered approaches. In the first configuration, the network is trained from scratch using random weights initialization. In the second one, learned parameters from the whole SC^2 corpus are used as starting point, but only the weights of the dense layers are adapted to the new task. In this transfer learning step, a small learning rate is used along with less training epochs. Concretely, for training the network from scratch a learning rate of 0.002 and 1900 epochs are considered, whilst for the network adaptation a learning rate of 0.0002 and 900 epochs have been tested. Since the network provides a continuous satisfaction value for each time window, different functions are compared to aggregate those values and provide a single satisfaction score for the whole conversation.

The results reported in Table 2 show that the aggregation based on the median value of the satisfaction estimates reaches the best performance, obtaining 0.60 AUC for the random initialization configuration and 0.584 AUC for the adapted network respectively. Table 4 details the performance of both networks on the satisfaction testing subset considering different audio segment duration and for the median aggregation. Results indicate that both systems obtain the best performance using as much data as possible, 4 minutes of each conversation, followed by the scores obtained considering only the last 30 seconds of the conversations. It is worth to note that the end-to-end system trained with no prior knowledge, *Random Initialization*, outperforms the proposed

feature transfer system. Nevertheless, in both cases, learned filters by the convolutional layers are useful again even in a harder task as detecting satisfaction relying on self-reported labels per the whole conversation.

4 Conclusions

The major contributions of our work are in the novel CNN architecture presented for extracting salient information directly from raw audio, together with the validation of such descriptors in a call center data set. Results reported on the SSPNet challenge data show that the proposed end-to-end architecture is able to learn features directly from structures in the time domain and having comparable results than those using classical Mel filter bank energies. Rather than that we show that the CNN based system performs comparable to the systems based on traditional hand-designed features and that the learned features are informative for the estimation of self-reported satisfaction.

5 Acknowledgements

We would like to thank the AVA innovation team members, among them, Roberto González and Nuria Oliver for interesting review. This project has received funding from the EU's Horizon 2020 research and innovation programme under grant agreement No 645323. This text reflects only the author's view and the Commission is not responsible for any use that may be made of the information it contains.

References

1. Abdel-Hamid, O., Mohamed, A.r., Jiang, H., Penn, G.: Applying convolutional neural networks concepts to hybrid nn-hmm model for speech recognition. In: Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on. pp. 4277–4280 (March 2012) 2
2. Bergstra, J., et al.: Theano: a cpu and gpu math expression compiler. In: Proceedings of the Python for scientific computing conference (SciPy). vol. 4, p. 3. Austin, TX (2010) 6
3. Budnik, M., Gutierrez-Gomez, E.L., Safadi, B., Quénot, G.: Learned features versus engineered features for semantic video indexing. In: Content-Based Multimedia Indexing (CBMI), 2015 13th International Workshop on. pp. 1–6 (June 2015) 2
4. Deng, L., Li, J., et al.: Recent advances in deep learning for speech research at Microsoft. In: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. pp. 8604–8608. IEEE (2013) 2
5. Devillers, L., Vaudable, C., Chastagnol, C.: Real-life emotion-related states detection in call centers: a cross-corpora study. In: Eleventh Annual Conference of the International Speech Communication Association. vol. 10, pp. 2350–2353 (2010) 2
6. Dieleman, S., Schrauwen, B.: End-to-end learning for music audio. In: Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on. pp. 6964–6968 (May 2014) 2, 5
7. Eyben, F., Wollmer, M., Schuller, B.: Openear - Introducing the Munich open-source emotion and affect recognition toolkit. In: Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009. 3rd International Conference on. pp. 1–6 (2009) 7
8. Goodfellow, I.J., Warde-Farley, D., Mirza, M., Courville, A.C., Bengio, Y.: Maxout networks. International Conference on Machine Learning (ICML) 28, 1319–1327 (2013) 4

9. Hinton, G., et al.: Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. Signal Proc. Magazine, IEEE 29(6), 82–97 (2012) 2
10. Hoshen, Y., Weiss, R.J., Wilson, K.W.: Speech acoustic modeling from raw multichannel waveforms. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 4624–4628. IEEE (2015) 2, 5
11. Huang, D.Y., Li, H., Dong, M.: Ensemble Nyström method for predicting conflict level from speech. In: Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific. pp. 1–5 (Dec 2014) 7
12. Jaitly, N., Hinton, G.: Learning a better representation of speech soundwaves using restricted Boltzmann machines. In: Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on. pp. 5884–5887. IEEE (2011) 2
13. Kim, S., Filippone, M., Valente, F., Vinciarelli, A.: Predicting the conflict level in television political debates: an approach based on crowdsourcing, nonverbal communication and gaussian processes. In: Proceedings of the 20th ACM international conference on Multimedia. pp. 793–796. ACM (2012) 6
14. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012) 3
15. Le, Q.V.: Building high-level features using large scale unsupervised learning. In: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. pp. 8595–8598 (May 2013) 2
16. LeCun, Y., Bengio, Y.: Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks 3361(10) (1995) 3
17. Llimona, Q., Luque, J., Anguera, X., Hidalgo, Z., Park, S., Oliver, N.: Effect of gender and call duration on customer satisfaction in call center big data. In: INTERSPEECH 2015 – 16th Annual Conference of the International Speech Communication Association, September 6–10, Dresden, Germany, Proceedings (2015) 6
18. Palaz, D., Magimai-Doss, M., Collobert, R.: Convolutional neural networks-based continuous speech recognition using raw speech signal. In: Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on. pp. 4295–4299 (April 2015) 2
19. Park, Y., Gates, S.C.: Towards real-time measurement of customer satisfaction using automatically generated call transcripts. In: Proceedings of the 18th ACM conference on Information and knowledge management. pp. 1387–1396. ACM (2009) 2
20. Räsänen, O., Pohjalainen, J.: Random subset feature selection in automatic recognition of developmental disorders, affective states, and level of conflict from speech. In: INTERSPEECH. pp. 210–214 (2013) 7
21. Schuller, B., et al.: The interspeech 2013 computational paralinguistics challenge: Social signals, conflict, emotion, autism 3, 7
22. Vaudable, C., Devillers, L.: Negative emotions detection as an indicator of dialogs quality in call centers. In: Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on. pp. 5109–5112. IEEE (2012) 2
23. Vinciarelli, A., Kim, S., Valente, F., Salamin, H.: Collecting data for socially intelligent surveillance and monitoring approaches: The case of conflict in competitive conversations. In: Communications Control and Signal Processing (ISCCSP), 2012 5th International Symposium on. pp. 1–4 (May 2012) 6
24. Zweig, G., Siohan, O., Saon, G., Ramabhadran, B., Povey, D., Mangu, L., Kingsbury, B.: Automated quality monitoring for call centers using speech and nlp technologies. In: Proceedings of the 2006 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology: companion volume: demonstrations. pp. 292–295. Association for Computational Linguistics (2006) 2