POSTERIOR CALIBRATION FOR MULTI-CLASS PARALINGUISTIC CLASSIFICATION

Gábor Gosztolya[*]

MTA-SZTE Research Group
on Artificial Intelligence
Szeged, Hungary

Róbert Busa-Fekete

Yahoo Research Inc.
New York, NY

ABSTRACT

Computational paralinguistics is an area which contains diverse classification tasks. In many cases the class distribution of these tasks is highly imbalanced by nature, as the phenomena needed to detect in human speech do not occur uniformly. To ignore this imbalance, it is common to measure the efficiency of classification approaches via the Unweighted Average Recall (UAR) metric in this area. However, general classification methods such as Support-Vector Machines (SVM) and Deep Neural Networks (DNNs) were shown to focus on traditional classification accuracy, which might lead to a suboptimal performance for imbalanced datasets. In this study we show that by performing posterior calibration, this effect can be countered and the UAR scores obtained might be improved. Our approach led to relative error reduction values of 4% and 14% on the test set of two multi-class paralinguistic datasets that had imbalanced class distributions, outperforming the traditional downsampling.

Index Terms— computational paralinguistics, classification, posterior estimates, posterior calibration

1. INTRODUCTION

Computational paralinguistics, a subfield of speech technology, focuses on extracting and predicting non-linguistic information present in human speech. Notable applications include the detection of laughter events [1, 2], emotion detection [3, 4], and various medical applications like early screening of Alzheimer's or Parkinson's disease [5, 6]. This area has many classification tasks; still, practically no effort is devoted for posterior probability calibration, which is a common technique in the machine learning literature (see e.g. [7, 8, 9]). The aim of our current study is to show that, simple posterior calibration technique is particularly beneficial in this area.

The goal of calibration is usually to turn the output scores of a classifier into valid posterior class probability estimates. It is usually employed when the learner method seeks to minimize quality measures such as the zero-one error, and this maximization does not demand precise posterior estimates. Examples for this are the output scores of AdaBoost.MH [10] or multi-class SVM [11]. This kind of inconsistency of output scores has *methodological* roots and various calibration techniques had been devised to handle it [12, 13]. Another situation where one might consider applying posterior calibration is when the classifier method applied seeks optimal *classification accuracy*, but we expect it to perform well with a different evaluation criterion such as optimal F-measure [14]. It is easy to see that this is just the case for several computational paralinguistic tasks, where classification performance is usually measured via the Unweighted Average Recall (UAR) metric, being the mean of the class-wise recall values. In cases where the class distribution is imbalanced, classifiers trained to maximize accuracy might provide suboptimal classification in terms of UAR scores.

There are some approaches which can be employed in such cases. One method of choice can be to use training examples belonging to the rarer classes multiple times (*upsampling*) or to discard examples belonging to the more frequent classes (*downsampling*) during training. However, upsampling leads to higher memory requirements, while downsampling might lead to information loss, as discarding training examples results in a degraded variance of the training data. Some approaches modify the training method itself either by employing more complicated sampling techniques such as probabilistic sampling [15] or instance re-weighting [16]; these, however, are not that straightforward to use in standard machine learning implementations.

We decided to apply a posterior calibration technique to adjust the output posterior estimates of a standard classifier in order to increase the UAR score of our predictions. This in practice means that we apply a well-known posterior calibration approach called *Platt scaling* [12]; the parameters of this method are tuned to achieve optimal UAR score. The motivation of this approach is that the optimal prediction for UAR depends on the fraction of the class conditional probabilities *and* class priors (see Section 2.2), whereas standard calibration techniques aim at improving only the class conditional probabilities.

Note that applying a calibration procedure is more impor-

[*] This study was supported by the National Research, Development and Innovation Office of Hungary via contract NKFIH FK-124413.

tant in multi-class setups than in two-class tasks. Assuming that the calibration process fits a monotonic function to the (raw) classifier outputs, in a binary problem calibration leads to the same results as classifier output thresholding. Since this is clearly not the case for multi-class problems, fitting more complex functions to the raw classifier outputs makes sense. Therefore, we will use two multi-class datasets in our experiments, both of which have a significantly inbalanced class distribution.

The structure of this paper is as follows. In Section 2, we introduce the calibration algorithm we used in our experiments (namely Platt scaling), explain our method of posterior re-calibration, and describe our approach of parameter setting. Then, in Section 3, we describe our experimental setup: the two datasets we perform our experiments on, the way we obtained our initial posterior estimates, and the technical details of posterior calibration. In Section 4, we present and analyze the test results; lastly, we draw our conclusions.

2. POSTERIOR CALIBRATION

2.1. Platt Scaling

Platt proposed using a sigmoid function to map the outputs of an SVM to posterior scores [12]. To get calibrated probability values in the binary case (i.e. the label is either zero or one) from some raw classifier output $\mathbf{f} : \mathbb{R}^d \mapsto \mathbb{R}$, we pass these output scores through a sigmoid function. That is, we have

$$P(y = 1|\mathbf{x}) = \frac{1}{1 + e^{a\mathbf{f}(\mathbf{x})+b}} (= s(\mathbf{f}(\mathbf{x}); a, b)), \qquad (1)$$

where the parameters a and b are fitted using maximum likelihood estimation on a calibration set. More concretely, a and b are solutions to the minimization problem

$$\operatorname*{argmin}_{a,b} - \sum_{i=1}^{n} y_i \log(p_i(a, b)) + (1 - y_i) \log(1 - p_i(a, b)), \qquad (2)$$

where $p_i(a, b) = s(\mathbf{f}(\mathbf{x}_i); a, b)$.

Given a multi-class dataset $\mathcal{D} = \{(\mathbf{x}_i, \boldsymbol{y}_i)\}_{i=1,\ldots,n}$, where $\boldsymbol{y}_i = (y_{i,1}, \ldots, y_{i,m})$ with exactly one non-zero label, one can also apply the likelihood principle to a raw multi-class classifier in the form of $\mathbf{f} : \mathbb{R}^d \mapsto \mathbb{R}^m$ to calibrate its output based on class-wise sigmoid functions. The straightforward generalization of Eq. (2) for the multi-class case is

$$\operatorname*{argmin}_{\mathbf{a},\mathbf{b}} - \sum_{i=1}^{n} \sum_{j=1}^{m} y_{i,j} \log(p_{i,j}(\mathbf{a}, \mathbf{b})), \qquad (3)$$

where $\mathbf{a} = (a_1, \ldots, a_m), \mathbf{b} = (b_1, \ldots, b_m)$ and

$$p_{i,j}(\mathbf{a}, \mathbf{b}) = \frac{s\big(f_j(\mathbf{x}_i); a_j, b_j\big)}{\sum_{\ell=1}^{m} s(f_\ell(\mathbf{x}_i); a_\ell, b_\ell)},$$

where $f_i(\mathbf{x})$ denotes the ith component of $\mathbf{f}(\mathbf{x})$. Note that $p_{i,j}(\mathbf{a}, \mathbf{b})$ is the estimate for the class conditional $P(y_j = 1|\mathbf{x}_i)$.

2.2. Optimal UAR

The Average Recall or Unweighted Accuracy (UAR) is the average of the classwise recalls. Assume a multi-class classifier in the form of $\mathbf{f} : \mathbb{R}^d \mapsto \{0, 1\}^m$ and a dataset $\mathcal{D} = \{(\mathbf{x}_i, \boldsymbol{y}_i)\}_{i=1,\ldots,n}$. Then the UAR score can be computed as

$$\text{UAR}(f, \mathcal{D}) = \frac{1}{m} \sum_{j=1}^{m} \frac{\sum_{i=1}^{n} y_{i,j} f_j(\mathbf{x}_i)}{\sum_{i=1}^{n} y_{i,j}}, \qquad (4)$$

where $f_j(\mathbf{x}_i)$ is the prediction of $\mathbf{f}(\mathbf{x}_i)$ for class j. One might be interested in the population level UAR for classifier $\mathbf{f}$ which can be written as

$$\mathbf{E}_{(\mathbf{x},\boldsymbol{y})\sim\mathbf{P}} [\text{UAR}(f, (\mathbf{x}, \boldsymbol{y})] = \frac{1}{m} \sum_{j=1}^{m} \frac{\mathbf{E}[y_j f_j(\mathbf{x})]}{\pi_j} \qquad (5)$$

where $\mathbf{E}[y_j f_j(\mathbf{x})]$ is the population level true positive rate of $\mathbf{f}$ for class j, and $\pi_i = \mathbf{P}(y_i = 1)$ is the prior for class j. Clearly, $\sum_{j=1}^{m} \pi_j = 1$ in a multi-class case. Because of the Law of large numbers, (4) converges to (5) almost surely.

Let us condition (5) on $\mathbf{x}$, which yields

$$\mathbf{E}_{\boldsymbol{y}\sim\mathbf{P}(.|\mathbf{x})} [\text{UAR}(f, \boldsymbol{y})] = \frac{1}{m} \sum_{j=1}^{m} \frac{\mathbf{E}[y_j|\mathbf{x}] f(\mathbf{x})_j}{\pi_j} \qquad (6)$$

By definition of conditional expectation, we have that

$$\mathbf{E}_{(\mathbf{x},\boldsymbol{y})\sim\mathbf{P}} [\text{UAR}(f, (\mathbf{x}, \boldsymbol{y})] = \mathbf{E}_{\mathbf{x}}[\mathbf{E}_{\boldsymbol{y}\sim\mathbf{P}(.|\mathbf{x})} [\text{UAR}(f, \boldsymbol{y})]],$$

where $\mathbf{E}_{\mathbf{x}}$ denotes the expectation with respect to the marginal distribution of the feature vectors.

The classifier $\mathbf{f}^*$ which maximizes (6) assigns a label

$$j^*_{\mathbf{x}} \in \operatorname*{argmax}_{j \in 1,\ldots,m} \frac{\mathbf{E}[y_j|\mathbf{x}]}{\pi_j}$$

to $\mathbf{x}$. In other words, for maximizing UAR, we need a fairly good estimate for the fraction of the class priors and the class conditional probabilities $\mathbf{E}[y_j|\mathbf{x}] = \mathbf{P}(y_i = 1|\mathbf{x})$ for each class. On the one hand, to have an accurate enough estimate for this fraction might be challenging for rare classes which occur typically for imbalanced dataset. On the other hand, the standard multi-class calibration given in (3) only aims at having accurate class conditional estimates, which is not sufficient in the case of UAR. This is why we devised a more robust calibration for UAR, which we will present next.

2.3. Posterior Re-Calibration

A classifier method provides posterior estimates which are usually already calibrated in some way. For example, when using a Deep Neural Network (DNN), it is standard practice to utilize neurons in the output layer which apply the softmax activation function; this already ensures that the output scores

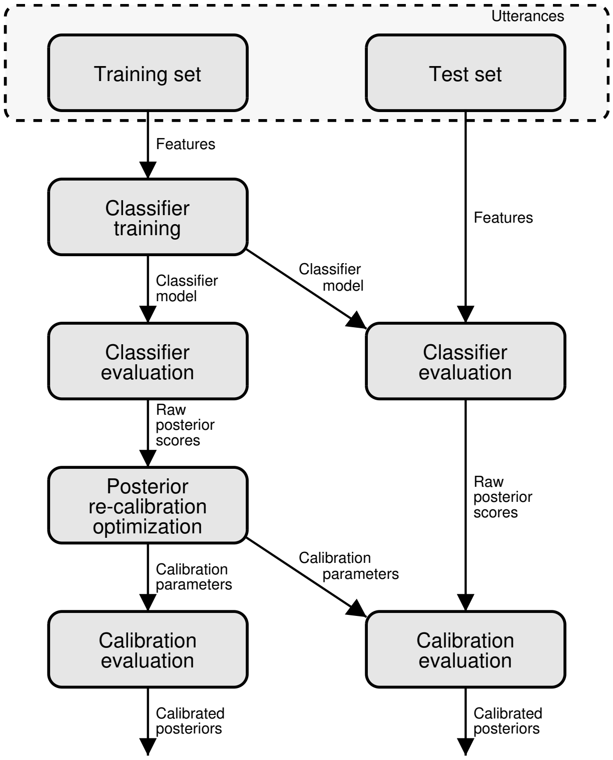

Fig. 1. The general workflow of the proposed paralinguistic posterior re-calibration process.

fall in the range $[0, 1]$ and add up to one, while for a Support-Vector Machine, Platt's method is normally applied. In our current study, we have posterior estimate scores as inputs, and our aim is only to re-calibrate these posterior scores to allow for a more balanced classification behaviour. Unfortunately, most classifier implementations do not offer access to the instance- and class-wise raw classifier output scores (i.e. the $f_i(\mathbf{x})$ values, where $\mathbf{f} : \mathbb{R}^d \mapsto \mathbb{R}^m$), therefore we have to find a way to perform posterior calibration from the already calibrated posterior estimate values. Notice, however, that by applying the inverse of the sigmoid function, we can obtain a linear function of the raw classifier output scores from the Platt-calibrated posterior estimates supplied by an SVM. That is,

$$a_i f_i(\mathbf{x}) + b_i = \log\Big(\frac{1}{p_i} - 1\Big), \tag{7}$$

a_i and b_i being the original calibration parameters. Denoting $a_i f_i(\mathbf{x}) + b_i$ by $f'_i(\mathbf{x})$, next we apply Platt's calibration function again, using the $\mathbf{f}'$ values as input and the new calibration parameters a'_i and b'_i. That is,

$$\begin{aligned} p'_i &= \frac{1}{1 + e^{a'_i f'_i(\mathbf{x}) + b'_i}} \\ &= \frac{1}{1 + e^{a'_i(a_i f_i(\mathbf{x}) + b_i) + b'_i}} \\ &= \frac{1}{1 + e^{(a'_i a_i) f_i(\mathbf{x}) + (a'_i b_i + b'_i)}}. \end{aligned} \tag{8}$$

Practically speaking, calibrating the f'_i values with Platt's formula (i.e. following Eq. (1)) leads to Platt-calibrated posterior estimates of the original, uncalibrated classifier output values (i.e. f_is), without the requirement of having direct access to these scores. We will exploit this finding in our experiments.

2.4. Obtaining Robust Calibration Parameters

Having the means of posterior re-calibration at our disposal, we still need to find a way to determine the a' and b' vectors which permit a higher-quality classification. A standard solution for such parameter optimization is to fine-tune these parameters on a development set or in a cross-validation setting [13, 9]. Translating this approach to our task, it means that we optimize the a' and b' vectors in order to get the highest UAR score; the robustness of this approach can be measured by transforming the posterior estimates of the test set using the found a' and b' parameter vectors. To find these parameters, we tested two such optimization approaches in our experiments.

To find these parameters, we tested two such optimization approaches in our experiments. In the first one, we generated random vectors and chose the one that led to the highest UAR value. Though this may seem to be a primitive technique at first glance, it was shown (see e.g. the study of Bergstra and Bengio [17]) that this is a favorable method to grid search for hyper-parameter optimization. The other optimization approach we applied is the **Covariance Matrix Adaptation Evolution Strategy** (CMA-ES, [18]) method. It is viewed as a reliable and competitive method for both local and global optimization [19]. . It has a further advantage that it requires little or no meta-parameter setting for optimal performance. We used the Java implementation with the default settings.

3. EXPERIMENTAL SETUP

3.1. The FAU AIBO Emotion Corpus

The FAU AIBO Emotion Corpus [20] contains audio files recorded from German children while playing with Sony's pet robot Aibo. The children were told that the Aibo responds to their commands, while it was actually remotely controlled by a human. Overall, 51 children were involved in the study from two schools; recordings from the Ohm school (9959 utterances) are commonly used as the training set in a speaker-wise cross-validation (CV) set-up, while data from the Mont school (8257 recordings) serve as the test set.

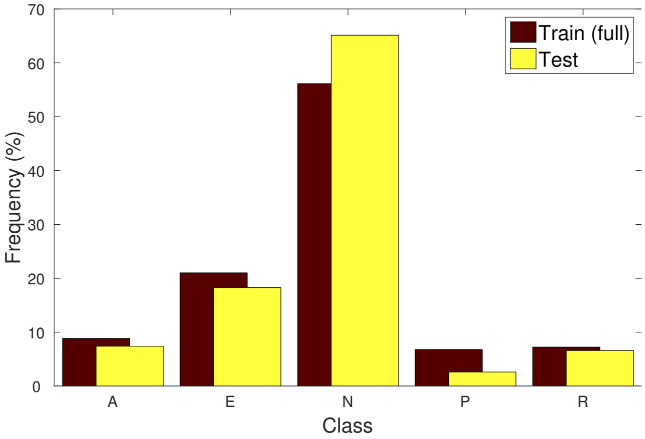

Fig. 2. The class distribution of the Emotion corpus.

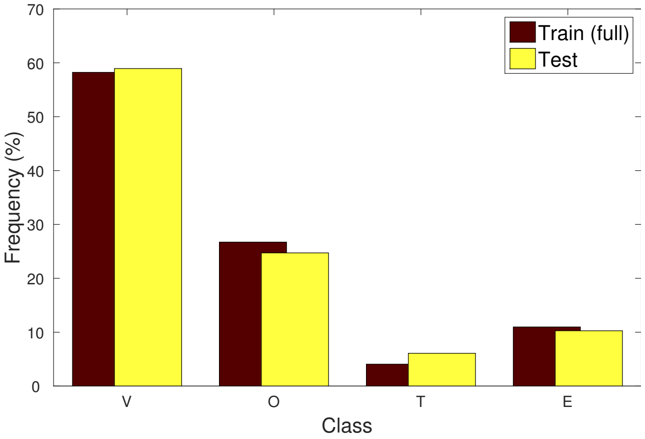

Fig. 3. The class distribution of the Snore corpus.

From the original 11 emotional categories, later a 5-class problem was created by merging emotional labels [21]. These classes are: Angry (*A*, containing the original categories of *angry*, *touch* and *reprimanding*), Emphatic (*E*), Neutral (*N*), Positive (*P*, containing *motherese* and *joyful*) and Rest (*R*). Fig. 2 shows the distribution of these five classes in the training and test sets. It is clear that class balance is quite uneven, most of the utterances belonging to the Neutral category (56% and 65%, training and test sets, respectively). This is understandable, though, since emotions do not have an equal distribution in everyday conversations either.

3.2. The Munich-Passau Snore Corpus

The Munich-Passau Snore Corpus [22, 23] contains sounds of 828 snoring events from 219 subjects. The dataset contains recordings of four types of snore events characterized based on the excitation location; there are Velum (*V*), Oropharyngeal lateral walls (*O*), Tongue (*T*) and Epiglottis (*E*) types of events [24], leading to four classification classes overall.

Although this corpus has a standard training, development and test split, we performed cross-validation on the combined training and development sets of this dataset as well. We did this as the number of recordings in this dataset is quite small, and using a separate development set would had halved the number of actual training examples. This would mean that the predictions would significantly differ for the development and test sets, because test set predictions are made by training a classifier on the training and development sets combined (i.e. roughly 600 recordings). Thus we performed ten-fold cross validation on the training and development sets combined, and we made sure that the snore sounds of each speaker appeared in one fold only.

The distribution of the examples belonging to each class in this corpus can be seen in Fig. 3. We can observe that class distribution is just as uneven for this corpus as it was for the Emotion corpus, 60% of the examples belonging to class *V*.

3.3. The Classification Process

Our classification process basically followed standard paralinguistic mechanisms (see e.g. [22, 25]): we used 6373 features overall, extracted by using the openSMILE tool [26]. The feature set includes energy, spectral, cepstral (MFCC) and voicing related low-level descriptors (LLDs), from which specific functionals (like the mean, standard deviation etc.) are computed to provide utterance-level feature values. After standardization, we trained a Support-Vector Machine, using the LibSVM [27] library. We used the nu-SVM method with a linear kernel; the value of C was tested in the range $10^{\{-5,\ldots,1\}}$, just like in our previous paralinguistic studies (e.g. [28, 29]).

3.4. Posterior Calibration

For both paralinguistic tasks, we optimized the a' and b' vectors of Eq. (8) on the posterior scores obtained on the whole training set in CV setup. This led to a 10-dimensional optimization task for the Emotion corpus and an 8-dimensional one for the Snore corpus. All the parameters were allowed to take values in the $[-10, 10]$ range. For the random optimization method, we generated 10,000 random vectors, while the CMA-ES method applied was allowed to optimize up to 10,000 iterations.

4. RESULTS

Table 1 contains the accuracy and UAR scores obtained on the **FAU AIBO Emotion corpus**. It is clear that the baseline approach had an uneven performance: the classification accuracy score of 64.5% is paired with a 33.3% UAR score. We also notice that class imbalance and the classifier method focusing on traditional classification accuracy were significant sources of the (relatively) low UAR scores, since downsampling led to higher UAR and lower accuracy scores. the train-

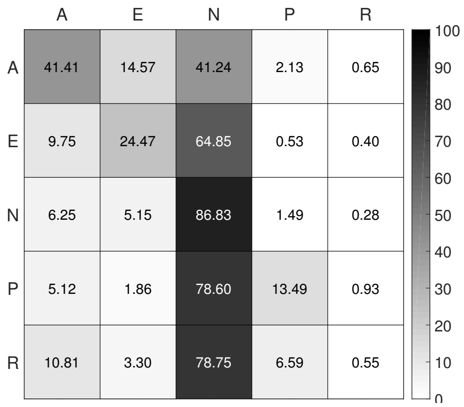

Fig. 4. Normalized confusion matrix of the test set of the FAU AIBO Emotion corpus using the original posterior values; the corresponding UAR score is 33.3%.

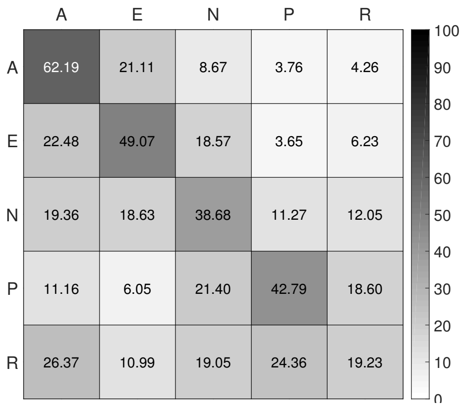

Fig. 5. Normalized confusion matrix of the test set of the FAU AIBO Emotion corpus using the re-calibrated posterior values; the corresponding UAR score is 42.4%.

Table 1. The accuracy and UAR scores obtained on the FAU AIBO Emotion corpus

	CV		**Test**	
Method	Acc.	UAR	Acc.	UAR
SVM (baseline)	62.5%	37.7%	64.5%	33.3%
Downsampling	62.7%	43.4%	35.9%	37.8%
Division by priors	47.9%	46.5%	42.1%	42.2%
Calibration (Random)	56.9%	43.1%	54.1%	38.9%
Calibration (CMA-ES)	44.7%	45.3%	41.1%	42.4%

Table 2. The accuracy and UAR scores obtained on the Munich-Passau Snore corpus

	CV		**Test**	
Method	Acc.	UAR	Acc.	UAR
SVM (baseline)	74.7%	57.2%	69.6%	53.9%
Downsampling	64.8%	57.6%	55.1%	45.9%
Upsampling	69.0%	55.0%	60.5%	47.3%
Calibration (Random)	75.8%	62.4%	69.2%	55.2%
Calibration (CMA-ES)	73.1%	64.3%	66.9%	55.8%

ing and test sets. (Note that we found upsampling to be impractical on this dataset, as it would have led to a huge number of training examples.)

Regarding posterior re-calibration, we can see that a large improvement can be achieved in the UAR values via this technique, even when we optimized the a' and b' parameter vectors via random value generation: the UAR values improved relatively via 9% and 8%, training and test sets, respectively. Using the CMA-ES optimization algorithm led to the even larger relative error reduction values of 12% and 14%. Notice that at the same time the accuracy scores decreased to the level of the UAR scores; on the test set, using the original posterior estimates produced a classification accuracy of 64.5%, which fell to 41.1% when we used the CMA-ES method for calibration parameter optimization. We, of course, do not consider it as a drawback, but an indication of a more balanced classification behaviour.

Examining the normalized confusion matrices got on the test set of this corpus by using the original posterior estimates (see Fig. 4) and the re-scaled ones after determining the parameters via CMA-ES (see Fig. 5), we can again notice this more balanced behaviour. In the former case, most examples were classified as neutral (N), being the most frequent class in the dataset. This led to a recall score of 87% for this class, but for the other classes we got much lower recall scores: for the R class this value was actually less than 1%. After posterior re-scaling, for most classes the recall scores lay between 35 and 50%, and even for the R class we got a recall score of over 19%.

Table 2 contains the accuracy and UAR scores obtained on the **Munich-Passau Snore Sound corpus**. We observe similar trends as we saw in the case of the emotion dataset: using the original posterior estimates led to relatively high classification accuracy and low UAR scores on both sets. Downsampling decreased both metric values by a significant amount. This is understandable, though, since the T class had only 23 examples in the full training set for this dataset, therefore downsampling led us to train a classifier model on just 92 examples overall. Upsampling surprisingly led to similar values. By posterior re-calibration, we were able to achieve large improvements in the cross-validation setting, but on the test set we got only 4% in terms of relative error reduction. Examining the confusion matrix, we found that we got low recall scores for the class T. Unfortunately, this class has only 16 instances in the test set, which makes it quite unreliable for measuring the actual performance of a classification process. In our opinion, however, this is a limitation of this particular dataset, also noted by Janott et al. [23]. However, the proposed posterior re-calibration approach still proved to be effective for improving the UAR scores on both multi-class computational paralinguistic datasets.

5. CONCLUSIONS

In computational paralinguistic classification tasks the de facto standard evaluation metric is Unweighted Average Recall; however, machine learning methods tend to optimize traditional accuracy. In this study we handled this inconsistency via posterior calibration. We showed that for optimal UAR, we can simply re-calibrate the posterior estimates from the original posterior and prior values; but this holds only if these values are sufficiently accurate, and this is unlikely for rarer classes. Therefore we performed a re-calibration of the original posterior estimates; we optimized the calibration parameters either by generating random values and by a robust optimization algorithm. Our approach led to relative error reduction values of 4% and 14% on the test set of two multi-class paralinguistic datasets that had imbalanced class distributions.

6. REFERENCES

[1] Lyndon S. Kennedy and Daniel P. W. Ellis, “Laughter detection in meetings,” in *Proceedings of the NIST Meeting Recognition Workshop at ICASSP*, Montreal, Canada, 2004, pp. 118–121.

[2] Gábor Gosztolya, András Beke, Tilda Neuberger, and László Tóth, “Laughter classification using Deep Rectifier Neural Networks with a minimal feature subset,” *Archives of Acoustics*, vol. 41, no. 4, pp. 1–10, 2016.

[3] Sz. L. Tóth, D. Sztahó, and K. Vicsi, “Speech emotion perception by human and machine,” in *Proceedings of COST Action*, Patras, Greece, 2012, pp. 213–224.

[4] Heysem Kaya and Alexey A. Karpov, “Efficient and effective strategies for cross-corpus acoustic emotion recognition,” *Neurocomputing*, vol. 275, pp. 1028–1034, 2018.

[5] Juan-Rafael Orozco-Arroyave, J.D. Arias-Londono, J.F. Vargas-Bonilla, and Elmar Nöth, “Analysis of speech from people with Parkinson’s disease through nonlinear dynamics,” in *Proceedings of NoLISP*, 2013, pp. 112–119.

[6] I. Hoffmann, D. Németh, C.D. Dye, M. Pákáski, T. Irinyi, and J Kálmán, “Temporal parameters of spontaneous speech in Alzheimer’s disease,” *International Journal of Speech-Language Pathology*, vol. 12, no. 1, pp. 29–34, 2010.

[7] Kristof Coussement and Wouter Buckinx, “A probability-mapping algorithm for calibrating the posterior probabilities: A direct marketing application,” *European Journal of Operational Research*, vol. 214, no. 3, pp. 732–738, 2011.

[8] Andrea Dal Pozzolo, Olivier Caelen, Reid A. Johnson, and Gianluca Bontempi, “Calibrating probability with undersampling for unbalanced classification,” in *Proceedings of CIDM*, Cape Town, South Africa, 2015, pp. 159–166.

[9] Khanh Nguyen and Brendan O’Connor, “Posterior calibration and exploratory analysis for natural language processing models,” in *Proceedings of EMNLP*, Lisbon, Portugal, Sep 2015, pp. 1587–1598.

[10] Robert E. Schapire and Yoram Singer, “Improved boosting algorithms using confidence-rated predictions,” *Machine Learning*, vol. 37, no. 3, pp. 297–336, 1999.

[11] K. Crammer and Y. Singer, “On the algorithmic implementation of multiclass kernel-based vector machines,” *Journal of Machine Learning Research*, vol. 2, pp. 265–292, 2001.

[12] J. Platt, “Probabilistic outputs for Support Vector Machines and comparison to regularized likelihood methods,” in *Advances in Large Margin Classifiers*, A.J. Smola, P. Bartlett, B. Schoelkopf, and D. Schuurmans, Eds., pp. 61–74. MIT Press, 2000.

[13] A. Niculescu-Mizil and R. Caruana, “Obtaining calibrated probabilities from boosting,” in *Proceedings of UAI*, 2005, pp. 413–420.

[14] Willem Waegeman, Krzysztof Dembczynski, Arkadiusz Jachnik, Weiwei Cheng, and Eyke Hüllermeier, “On the Bayes-optimality of F-measure maximizers,” *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3333–3388, 2014.

[15] Tamás Grósz, Gábor Gosztolya, and László Tóth, “Training context-dependent DNN acoustic models using probabilistic sampling,” in *Proceedings of Interspeech*, Stockholm, Sweden, Aug 2017, pp. 1621–1625.

[16] Heysem Kaya and Alexey A. Karpov, “Introducing weighted kernel classifiers for handling imbalanced paralinguistic corpora: Snoring, addressee and cold,” in *Proceedings of Interspeech*, Stockholm, Sweden, Aug 2017, pp. 3527–3531.

[17] James Bergstra and Yoshua Bengio, “Random search for hyper-parameter optimization,” *Journal of Machine Learning Research*, vol. 13, pp. 281–305, 2012.

[18] N. Hansen and A. Ostermeier, “Completely derandomized self-adaptation in evolution strategies,” *Evolutionary Computation*, vol. 9, no. 2, pp. 159–195, 2001.

[19] N. Hansen and S. Kern, "Evaluating the CMA evolution strategy on multimodal test functions," in *Parallel Problem Solving from Nature PPSN VIII*, X. Yao et al., Eds. 2004, vol. 3242 of *LNCS*, pp. 282–291, Springer.

[20] Stefan Steidl, *Automatic Classification of Emotion-Related User States in Spontaneous Children's Speech*, Logos Verlag, Berlin, 2009.

[21] Björn Schuller, Stefan Steidl, and Anton Batliner, "The INTERSPEECH 2009 emotion challenge," in *Proceedings of Interspeech*, 2009, pp. 312–315.

[22] Björn Schuller, Stefan Steidl, Anton Batliner, Simone Hantke, Elika Bergelson, Jarek Krajewski, Christoph Janott, Andrei Amatuni, Marisa Casillas, Amanda Seidl, Melanie Soderstrom, Anne S. Warlaumont, Guillermo Hidalgo, Sebastian Schnieder, Clemens Heiser, Winfried Hohenhorst, Michael Herzog, Maximilian Schmitt, Kun Qian, Yue Zhang, George Trigeorgis, Panagiotis Tzirakis, and Stefanos Zafeiriou, "The INTERSPEECH 2017 computational paralinguistics challenge: Addressee, Cold & Snoring," in *Proceedings of Interspeech*, 2017, pp. 3442–3446.

[23] Christoph Janott, Maximilian Schmitt, Yue Zhang, Kun Qian, Vedhas Pandit, Zixing Zhang, Clemens Heiser, Winfried Hohenhorst, Michael Herzog, Werner Hemmert, and Björn Schuller, "Snoring classified: The Munich-Passau snore sound corpus," *Computers in Biology and Medicine*, vol. 94, no. 1, pp. 106–118, 2018.

[24] E.J. Kezirian, W. Hohenhorst, and N. de Vries, "Drug-induced sleep endoscopy: the VOTE classification," *Archives of Oto-Rhino-Laryngology*, vol. 268, no. 8, pp. 1233–1236, 2011.

[25] Björn Schuller, Stefan Steidl, Anton Batliner, Simone Hantke, Florian Hönig, Juan Rafael Orozco-Arroyave, Elmar Nöth, Yue Zhang, and Felix Weninger, "The INTERSPEECH 2015 computational paralinguistics challenge: Nativeness, Parkinson's & eating condition," in *Proceedings of Interspeech*, 2015, pp. 478–482.

[26] F. Eyben, M. Wöllmer, and B. Schuller, "Opensmile: The Munich versatile and fast open-source audio feature extractor," in *Proceedings of ACM Multimedia*, 2010, pp. 1459–1462.

[27] Chih-Chung Chang and Chih-Jeh Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 1–27, 2011.

[28] Gábor Gosztolya, Tamás Grósz, György Szaszák, and László Tóth, "Estimating the sincerity of apologies in speech by DNN rank learning and prosodic analysis," in *Proceedings of Interspeech*, San Francisco, CA, USA, Sep 2016, pp. 2026–2030.

[29] Gábor Gosztolya, Róbert Busa-Fekete, Tamás Grósz, and László Tóth, "DNN-based feature extraction and classifier combination for child-directed speech, cold and snoring identification," in *Proceedings of Interspeech*, Stockholm, Sweden, Aug 2017, pp. 3522–3526.