Age Group Classification with Speech and Metadata Multimodality Fusion

Denys Katerenchuk[*]
CUNY Graduate Center
365 Fifth Avenue, Room 4319
New York, USA
`dkaterenchuk@gradcenter.cuny.edu`

Abstract

Children comprise a significant proportion of TV viewers and it is worthwhile to customize the experience for them. However, identifying who is a child in the audience can be a challenging task. We present initial studies of a novel method which combines utterances with user metadata. In particular, we develop an ensemble of different machine learning techniques on different subsets of data to improve child detection. Our initial results show an 9.2% absolute improvement over the baseline, leading to a state-of-the-art performance.

1 Introduction

Building on recent breakthroughs on speech understanding, people ask their cellphones any questions and expect to get reasonable answers, or ask their TVs for movie recommendations. The identity of the user plays a key role in personalizing and improving these actions. For instance, in the case of movie request, a general, probabilistic model will not work well. Consider a case when a child asks to watch "Ruby and Max," an animated television series, but the automatic speech recognition system (ASR) mistakenly resolves it to the popular "Mad Max" movie in the downstream natural language processing (NLP) module. Having the knowledge of the age, the system could fix such errors by returning age-relevant results.

Unfortunately, this scenario is quite common considering that even state-of-the-art ASR systems produce very bad results on understanding children's speech. There are a couple reasons for

this: 1) most ASR systems are trained to understand adults, 2) children's voices are hard to analyze because of not fully developed vocal tracts (Shivakumar et al., 2014). One way to improve the performance is to add an intermediate system that can identify users.

In this paper we investigate child identification from voice commands, metadata and the combination of the two to improve classification accuracy. Age and gender identification from speech is not a new problem and much research has been done in this area (sec.2), yet the results are far from perfect. In particular, the task to identify adults from kids becomes more challenging when the utterances are only a couple of seconds long. We investigate a novel multimodel approach to improve classifier accuracy by combining speech data with rich usage metadata (sec.3). Specifically, we extract features separately from speech and usage data, and build individual models that are fused together (sec.4) to improve classifier performance. The results are described in section 5.

2 Related Work

Speaker information, such as accent, gender or age, can be used to improve speech understanding (Abdulla et al., 2001), provide background information, and advance human-computer interactions. A human vocal tract undergoes changes starting from birth and continues throughout one's life. Brown et al. (1991) found that fundamental frequencies directly correspond to the ages of professional singers. Later, Naini and Homayounpour (2006) investigated the correspondence of MFCCs, shimmer, and jitter to a speaker's age. They found that jitter and shimmer do, indeed, help distinguish ages, but only on wider age rages. With application of more advanced machine learning techniques, Metze et al. (2007) achieved hu-

---

man level performance on longer speech segments, while short utterances were challenging to classify correctly. The recent work on a similar task of gender identification by Levitan et al. (2016a), revealed that human level performance is achievable on short utterances as well.

In this work, we build on the prominent research approach, and investigate its performance on a challenging real world data set: TV domain where utterances are only about a second long. This is why in addition to speech, we analyze metadata, which is commonly ignored, to explore a fusion of multiple models in the classification task. We compare the performance of three models based on SVM, random forest, and deep learning, then report the results.

3 Data

The speech data is collected at random each week for over a year's time span and was manually labeled by human annotators as "MALE", "FEMALE", or "KID". Since we don't have ground truth labels, we use these labels as the gold standard. "MALE" and "FEMALE" labels are combined into one "ADULT" class. Each audio is a short, on average 1.2 seconds long, command from a user to a TV box such as "watch Sponge-Bob" or "CNN." In total we have 15,001 instances of labeled utterances where 3,848 were labeled as "KID". To normalize the data set, we at random sampled 3,848 utterances with the "ADULT" label. This is done to create a balanced dataset of 7,696 instances. The data was split into train and test sets with 75:25 ratio leaving 5,772 for training and 1,924 instances for testing sets. Cross validation on the train set was used as our development set to optimize the algorithms. The final results are reported on the test set.

In addition to the voice commands, we collected user metadata. This data contains general usage patterns such as date, time, and expected audience type (children or adults) of the requested TV show. The data covers only one month of activity which makes the data meager. As a result, we ignore dates and use weekdays instead. Additionally, we calculate the likelihood of a request made for a children's show in a given weekday and hour. All the times and date were converted to the user local time zones. In addition, we use a hand written rule that treats all commands between 11pm and 6am as commands from adults. The reason is that most

of the time children are in bed during these hours and in way we are eliminating false positives.

4 Methods

4.1 Feature Extraction

Before the feature extraction step, the audio was preprocessed and normalized. In the preprocessing step, all silences were removed to keep user commands. In the normalization step, we try to mitigate variance in speech by normalizing the volume. This is a common preprocessing step that is used in ASR systems. After these two preprocessing steps, we extract features to use as an input to train an acoustic model. In order to validate the quality of the preprocessing steps, acoustic features are extracted from raw, preprocessed, and normalized audio.

For acoustic feature extraction we use the open-source tool openSMILE (Eyben et al., 2010). OpenSMILE is a well known utility that produces state-of-the-art acoustic features and often used during annual INTERSPEECH paralinguistic challenges to define a baseline. The source code includes a set of configuration files for different features. The configuration file we use in our experiments is "paraling_IS10.conf." This version was introduced during the INTERSPEECH 2010 Paralinguistic Challenge (Schuller et al., 2010). The challenge was to create predictive models for gender and age classification. We also tried to experiment with other configuration files; however, showed lower performance.

We extract 1582 acoustic features from each user utterance. The features are created by first extracting low level descriptors (LLDs) of 10ms frame level step and 20ms window size. The LLDs include a total of 34 features such as 12 MFCCs, F0, energy, jitter, etc. After that, we derive 34 deltas from the LLDs and apply a set of 21 functions. A list of the functions is shown in table 1 and complete feature description can be found in (Schuller et al., 2010).

In addition to speech, we explore the user requests. Despite ever-changing TV content, some phrases or words can aid to identify the viewer's age. We use an ASR system on each utterance to extract a transcript. Since the commands are very short and specific to the domain, a simple bag-of-word language model (Zhang et al., 2010) is sufficient. From the dictionary of 5092 unique words, 2000 of the most frequent words are used

LLDs	Functions
mfcc 0-14	mean/max/min Pos
pcm_loudness	linregc 1 2
logMelFreqBand 0-7	linregrr A Q
lspFreq from 8 LPC	stddev, kurtosis
F0finEnv	quartile 1,2,3
voicingFinalUnclipped	percentile 1, 99
F0	prtl_range 0,1
jitter L/DDP	
shimmer	

Table 1: Acoustic features

as a word feature vector.

For each utterance, we also use its metadata such as weekday and hour. The intuition for including this data is that some TV content is targeted for a particular audience with respect to the time of the day. For example, news tend to run during evening hours and children oriented shows are shown in the morning or during a day.

In addition, we use the show-type request distribution from a given device. The distribution is derived by computing the percentage of children's shows against all shows watched during the specific time. We derive this distribution as a score from 0 to 1 for each hour, day, and entire one month time period for a given device. If a command comes between 11pm and 6am, we mark it with 0, assuming that only adults can be awake during these hours. As a result, two feature datasets were created: 1) time usage data that contains usage frequency per hour and weekday as well as content type, 2) ratio usage data that includes distribution of kids and non-kids content requested from a given device. The ratio is calculated for each hour, day, and a given device in general. Considering that usage patterns of users with and without children vary, these datasets will add important information to make better classification decisions.

4.2 Classification

For classification, we use two well known algorithms: support vector machines (SVM) (Suykens and Vandewalle, 1999) and random forest (Liaw and Wiener, 2002). Both algorithms show state-of-the-art performances in speaker classification tasks (Ahmad et al., 2015). SVM algorithm works by creating support vectors that separating two classes in n-dimensional space, where each dimen-

sion is represented by a feature. The separation is done by finding the largest separation margin between the features from the two classes and a vector. Random forest is a tree based ensemble algorithm that work by running multiple decision tree algorithms, which are known as weak learners, at the same time. Each algorithm at random selects features and makes its decisions. At the end, all the results from the each learner are combined to provide the prediction. The models are trained using scikit-learn toolkit (Pedregosa et al., 2011), an open-source machine learning library. Both algorithms were used for training. However, only the best algorithm is used on the test data.

Deep learning (LeCun et al., 2015) has shown to be a useful technique in many areas including audio processing. We build a deep network classifier with four hidden layers. Each layer is fully connected with a 50% dropout rate to reduce overfitting (Srivastava et al., 2014) and a sigmoid activation function (Marreiros et al., 2008). The last layer uses softmax activation and 0% dropout rate. The size of each layer is chosen to first generalize the features and then narrow the feature space size. The best architecture has the following layer sizes [1582, 1582*8, 2048, 512, 64, 2]. The first and last layers are acoustic feature input and predicted binary class output respectively. The network is trained overnight on a consumer level GPU.

During the training, we start with audio preprocessing by applying energy normalization and silence removal techniques. While energy normalization is a useful method to improve ASR performance (Li et al., 2001), the results need to be tested to determine if this approach is applicable to our task. The removal of silences, on the other hand, is a valid step to increase the accuracy. After determining the best audio normalization, we train a separate model for each feature set: 1) audio, 2) time usage data, and 3) show-type request distribution. The models are tested with cross validation and the scores are reported in section 5. The test sets are used only at the end to evaluate the models on previously unseen data. In this way we avoid overfitting by tuning the algorithms on train data with cross validation that we use as development dataset. At this point we have three datasets, which are acoustic data, time usage, and content type ratio. Each model is evaluated separately on the corresponding dataset.

Leveraging multi-domain data, we apply feature

Classifiers	WN	EN	SR
SVM	81.7%	79.8%	84.4%
Rand. Forest	81.3%	80.5%	86.7%

Table 2: Audio normalization of three subsets: WN - without normalization, EN - energy normalized, SR - silence removed.

Classifiers	Time	Show-type	BOW
SVM	53.4%	59.9%	64.7%
Rand. Forest	54.9%	56.8%	68.2%

Table 3: Training results on meta data

Audio	DL	Time	Show-type	BOW
86.6%	88.82	57.9%	55.8%	67.6%

Table 4: Test results

and model level fusion methods. We experiment with combining features from the three domains into a single feature vector and train additional model on these features. At the same time, we perform model level fusion (Huang et al., 2011). Each trained model's output probability is used as inputs to AdaBoost ensemble learning algorithm (Rätsch et al., 2001). We apply this approach only on the test data. The evaluation is done using cross validation on the test dataset.

5 Results

5.1 Baseline

For the baseline, we use INTERSPEECH 2010 paralinguistic gender and age challenge's pipeline (Schuller et al., 2010). The data that was used for the challenge is different from ours in terms of audio domain, quality, and utterances were on average 2.2% longer. Longer audio segment provide more information making the task easier. Furthermore, the challenge was to classify users of 4 age groups while we perform binary classification. For this reason we cannot directly compare the scores. However, we follow the steps to replicate the challenge's pipeline on our unaltered data and use the score as our baseline. The accuracy of the baseline is defined at 81.7%.

5.2 Training results

The first step is to choose the best normalization approach. We create three subsets of audio: without normalization (WN), energy level normalized (EN), and silence removed (SR) utterances. In order to find which technique works the best, we apply SVM and random forest to each subset. The results are shown in table 2.

From the table we can see that energy holds important information about the speaker and normalizing it worsens the predictions. In contrary, removing silences significantly improves the results in both classifiers. For this reason, we keep the silence removal preprocessing step in our pipeline and omit energy normalization. In addition, the

random forest outperforms the SVM algorithm in the majority of cases and confirms the results of (Levitan et al., 2016a; Levitan et al., 2016b) on similar tasks. Random forest will be used in the rest of our experiments as the main algorithm for utterance classification.

Metadata and language features were also tested with both SVM and random forest algorithms. Each algorithm is applied to three datasets 1) time usage data, 2) show-type request distribution, and 3) language bag-of-word (BOW) features. The performance is described in table 3.

We can see that time usage and show-type ratio provide very little information on who the user is. Bag-of-word model shows a prediction accuracy of 68.2%. This result better compares to metadata, but is worse than acoustic features alone. Random forest outperforms SVM on two out of three domain of the data. For this reason, we use random forest as out main machine learning algorithm on this data. All the experiments are tested by means of cross validation on training set that we use as our development set. Due to the time complexity of deep learning algorithm, we do not use it during cross validation. Having decided on the best normalization and machine learning algorithm, we are ready to see the performance on the test data set.

5.3 Testing results

The results on the test data are comparable to what we got during the cross validation on our train set. Table 4 shows that the time based model provides only 57.9% accuracy. The expectation was to get a higher score on this data set. Our hypothesis was that TV content providers use time slots to target different age groups of their audience. Weekend mornings for animated shows and weekday nights for news are examples of such. One reason for this might be that the commands for children shows come from parents. We also explored

	Baseline	Feature	Model
Accuracy	81.7%	86.3%	90.9%

Table 5: Feature and model level fusion results

	Adult	Kid
Adult	88%	7%
Kid	12%	93%

Table 6: Class confusion matrix

show-type requests for each device to capture user interest. This turned out to be the least predictive data model. The idea was that users with children will request more child oriented content. However, insufficient data size of our show-type distribution can be attributed to such low result, and larger data set may improve the performance. This will need further investigation. Acoustic based models are the most predictive. While random forest shows improved results compared to the baseline, the deep learning method outperformed all the models and showed 88.82% accuracy.

5.4 Feature and Model Fusion

We explore feature level and model level fusion approaches to improve the results. Both techniques are known ways to combine multi-domain data. We concatenated features from all four data sets and trained a new random forest model. This produced somewhat of an unpredictable result. The accuracy of the model did not improve, and even worsened producing 86.3% accuracy. It seems that combining all the available features into a single vector introduces noise and data sparsity problem. The acoustic model alone outperforms the feature lever fusion approach.

For our model level fusion approach, we use ensemble algorithm AdaBoost (Freund and Schapire, 1997). AdaBoost is an adaptive model that iteratively boosts weak learner to focus on harder cases in the training dataset. The input to this model are class probabilities from each of the five models, which are 1) random forest based acoustic, 2) time usage, 3) show-type requests, 4) bag-of-words language model, and 5)

	Male	Female	Kid
Adult	49%	39%	7%
Kid	1%	11%	93%

Table 7: Gender confusion matrix

deep learning based acoustic model. The results achieved by this approach produce 90.9% accuracy (table 5) manifesting in 9.2% absolute improvement. With closer investigation of the results in table 6, we can see that the algorithm works better to identify children with only 7% on false positives. However, the model produces higher error predicting adult voices as children. Table 7 shows gender based confusion matrix. From this table we can observe that the algorithm makes the most error distinguishing female from children voices. This comes from the fact that female voices have broader acoustic range compare to male and, as a result, they overlap with children's.

6 Conclusion

This work is focused on improving child and adult user classification from voice and metadata. Metadata provides additional information about user such as time, show-categories, and show-type distribution. This type of data is often ignored during research. We found that multi-domain feature level fusion did not help to improve the results. However, by combining the models using ensemble model fusion improves the performance. The system achieves 90.9% accuracy on the task and produces state-of-the-art results.

7 Future work

In our future work, we would like work to improve our model by investigating and capturing acoustic differences between female and child voices, since our current system produces the most error classifying these groups. Also, we would like to compare the performance of human engineered features and deep learning based feature representation. In addition, semi-supervised approaches have gain popularity. Data labeling is a costly and time consuming process that, for this problem, requires human annotators. Leveraging the large amount of unlabeled data can improve results even further.

References

W.H. Abdulla, N.K. Kasabov, and Dunedin-New Zealand. 2001. Improving speech recognition performance through gender separation. *changes*, 9:10.

Jamil Ahmad, Mustansar Fiaz, Soon il Kwon, Maleerat Sodanil, Bay Vo, and Sung Wook Baik. 2015. Gender identification using mfcc for telephone applications - a comparative study. *CoRR*, abs/1601.01577.

W.S. Brown, Richard J. Morris, Harry Hollien, and Elizabeth Howell. 1991. Speaking fundamental frequency characteristics as a function of age and professional singing. *Journal of Voice*, 5(4):310–315.

Florian Eyben, Martin Wöllmer, and Björn Schuller. 2010. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1459–1462. ACM.

Yoav Freund and Robert E. Schapire. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.*, 55(1):119–139, August.

Dong-Yan Huang, Shuzhi Sam Ge, and Zhengchen Zhang. 2011. Speaker state classification based on fusion of asymmetric simpls and support vector machines. In *INTERSPEECH*, pages 3301–3304.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature*, 521(7553):436–444.

Sarah Ita Levitan, Yocheved Levitan, Guozhen An, Michelle Levine, Rivka Levitan, Andrew Rosenberg, and Julia Hirschberg. 2016a. Identifying individual differences in gender, ethnicity, and personality from dialogue for deception detection. In *Proceedings of NAACL-HLT*, pages 40–44.

Sarah Ita Levitan, Taniya Mishra, and Srinivas Bangalore. 2016b. Automatic identification of gender from speech. In *Speech Prosody*.

Qi Li, Jinsong Zheng, Qiru Zhou, and Chin-Hui Lee. 2001. Robust, real-time endpoint detector with energy normalization for asr in adverse environments. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, volume 1, pages 233–236. IEEE.

Andy Liaw and Matthew Wiener. 2002. Classification and regression by randomforest. *R news*, 2(3):18–22.

André C. Marreiros, Jean Daunizeau, Stefan J. Kiebel, and Karl J. Friston. 2008. Population dynamics: variance and the sigmoid activation function. *Neuroimage*, 42(1):147–157.

Florian Metze, Jitendra Ajmera, Roman Englert, Udo Bub, Felix Burkhardt, Joachim Stegmann, Christian Muller, Richard Huber, Bernt Andrassy, Josef G Bauer, et al. 2007. Comparison of four approaches to age and gender recognition for telephone applications. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, volume 4, pages IV–1089. IEEE.

A. Sadeghi Naini and M.M. Homayounpour. 2006. Speaker age interval and sex identification based on jitters, shimmers and mean mfcc using supervised and unsupervised discriminative classification methods. In *2006 8th international Conference on Signal Processing*, volume 1. IEEE.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Gunnar Rätsch, Takashi Onoda, and K.-R. Müller. 2001. Soft margins for adaboost. *Machine learning*, 42(3):287–320.

Björn Schuller, Stefan Steidl, Anton Batliner, Felix Burkhardt, Laurence Devillers, Christian A. Müller, Shrikanth S. Narayanan, et al. 2010. The interspeech 2010 paralinguistic challenge. In *Inter-Speech*, volume 2010, pages 2795–2798.

Prashanth Gurunath Shivakumar, Alexandros Potamianos, Sungbok Lee, and Shrikanth Narayanan. 2014. Improving speech recognition for children using acoustic adaptation and pronunciation modeling. In *Proc. Workshop on Child, Computer and Interaction (WOCCI)*.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Johan A.K. Suykens and Joos Vandewalle. 1999. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300.

Yin Zhang, Rong Jin, and Zhi-Hua Zhou. 2010. Understanding bag-of-words model: a statistical framework. *International Journal of Machine Learning and Cybernetics*, 1(1-4):43–52.