Acoustic Scene Recognition with Deep Neural Networks (DCASE challenge 2016)

Dai Wei, Juncheng Li, Phuong Pham, Samarjit Das, Shuhui Qu,[·]

Robert Bosch Research and Technology Center,
Carnegie Mellon University,
Stanford University,
University of Pittsburgh

ABSTRACT

Background: Sounds carry a large amount of information about our everyday environment and physical events that take place in it. Complementing visual inputs, sound can be more easily collected and stored. Increasingly machines in various environments can hear, such as smartphones, autonomous robots, or security systems. This work applies state-of-the-art deep learning models that have revolutionized speech recognition to understanding general environmental sounds.

Goal: This work aims to discriminatively characterize sound in 15 common indoor and outdoor acoustic scenes by classifying audio recordings.

Data: We use dataset from the ongoing IEEE challenge on Detection and Classification of Acoustic Scenes and Events (DCASE). The dataset was collected in Finland by Tampere University of Technology between 06/2015 - 01/2016. It contains 15 diverse indoor and outdoor locations (classes), such as bus, cafe, car, city center, forest path, library, train, totaling 9.75 hours of audio recording.

Methods: We extract 4 sets of features using signal processing techniques, such as Mel-frequency cepstral coefficients (MFCC), various statistical functions, and spectrogram. The 4 features sets are: MFCC (60-dim), Smile983 (983-dim), and Smile6k (6573-dim). On these features we apply 5 models: Gaussian mixture model (GMM), Support Vector Machine (SVM), Deep Neural Network (DNN), Hierarchical DNN, Recurrent Neural Network (RNN), Recurrent Deep Neural Network(RDNN). Among them GMM and SVM are popular model for this task, while RDNN, is, to our knowledge, the first application of these models in the context of environmental sound.

Results: Model performance varies with features. With small set of features (MFCC and Smile983) temporal models (RNN, RDNN) outperform non-temporal models (GMM, SVM, DNN). However, with large feature sets (Smile6k) DNN outperforms temporal models (RNN and RDNN) and achieves the best performance among all studied methods.
GMM with MFCC feature, the baseline model provided by the DCASE contest, achieves 67.6% test accuracy, while the best performing model (hierarchical DNN model with Smile6k feature) reaches 82.3% test accuracy. RNN and RDNN generally have performance in the range of 68_77%, while SVM varies between 56_73%.

Conclusions: We find that deep learning models compare favorably to traditional models (GMM and SVM). No single model outperforms all the other models across all feature sets, showing that model performance varies significantly with feature representation. The fact that the best performing model is the non-temporal DNN model is an evidence that environmental scene sounds don't necessarily exhibit strong temporal dynamics. This is consistent with our day-to-day experience that environmental sounds tend to be random an unpredictable.

1. INTRODUCTION

Recent developments in deep learning has brought significant improvements to automatic speech recognition (ASR) (Hannun et al. (2014)) and music characterization (Van den Oord et al. (2013)). However, speech is only one of many types of sounds, and in practice, humans often rely on a broad range of environmental sounds to detect danger and enhance scene understanding, such as when one crosses a busy street or navigate in a bustling office. More broadly, sound is a useful modality complementing visual information such as videos and images, with the advantage that audio can be more easily collected and stored. Audio is also perspective and illumination invariant unlike its visual counterpart.

Increasingly, machines in various environments can hear, such as smartphones, security systems, and autonomous robots. The prospect of human-like sound understanding could open up a range of applications, including intelligent machine state monitoring using acoustic information, acoustic surveillance, cataloging and information retrieval applications such as search in audio archives (Ranft (2004)) as well as audio-assisted video/multimedia content search.

[·] Thanks to ABC agency for funding.

These broad range of environmental sounds also pose different challenges than speech recognition problems. Compared to speech, environmental sounds are more diverse and span a wide range of frequency. Moreover, they are often less well defined. For example, there's no standard dictionary for environmental events analogous to sub-word dictionary phonemes in speech, and environmental sounds' duration could vary widely. While sound analysis traditionally falls within signal processing domain, recent advances in machine learning and deep learning holds the promise to improves upon existing signal processing methods. In this work we focus on the task of acoustic scene identification, which aims to characterize the acoustic environment of an audio stream by selecting a semantic label for it. Existing works for this task largely use conventional classifiers such as GMM and SVM, which do not have the feature abstraction capability found in deeper models. Furthermore, conventional models do not model temporal dynamics, but rely on feature extraction pipeline to capture local temporal dynamics. For example, the winning solution by Roma et al. (2013) for an acoustic scene classification challenge in 2013 (the previous run of the current DCASE challenge), extracts MFCC and temporal features using Recurrence Quantification Analysis over a short time window. The actual classifier SVM does not explicitly model temporal dynamics.

We apply state-of-the-art deep learning (DL) architectures to various feature representations generated from signal processing methods. Specifically, we use the following architectures: (1) Deep Neural Network (DNN) and hierarchical DNN (2) Recurrent Neural Network (RNN); (3) Recurrent Deep Neural Network (RDNN. Additionally we compare DL models with Gaussian mixture model (GMM), and Support Vector Machine (SVM). We also use several feature representations based on signal processing techniques: Mel-frequency cepstral coefficients (MFCC), spectrogram, other conventional features such as pitch, energy, zero-crossing rate, mean-crossing rate etc. There are several studies using deep learning in sound event detection(Emre, 2015, Mesaros 2016). However, to the best of our knowledge, this is the first comprehensive study of a diverse set of deep architectures on acoustic scene recognition task, borrowing ideas from signal processing as well as recent advancements in automatic speech recognition.

We use a dataset from the currently ongoing IEEE challenge on Detection and Classification of Acoustic Scenes and Events (DCASE). The dataset contains 15 diverse indoor and outdoor locations (classes), such as bus, cafe, car, city center, forest path, library, train, totaling 9.75 hours of audio recording (see section 3.1 for detail). Our system has entered the DCASE 2016 contest, which runs from February 8 to June 30, 2016, at which point the final ranking of the contestants would be announced.

2. EXPERIMENT

2.1. Dataset

We use a dataset from the currently ongoing IEEE challenge on Detection and Classification of Acoustic Scenes and Events (Mesaros (2016)). The dataset contains 15 diverse indoor and outdoor locations (labels), totaling 9.75 hours of recording and 8.7GB in wav format. (Dual Channels, Sample Rate: 44100 Hz, Precision: 24 bit, Duration: 30 sec each)
The classes and key content tags (acquired through selectively listen to samples of the recordings) are listed as below:

Classes	**Key Content**
Bus: traveling by bus in the city (vehicle)	#low frequency noise; #vibration; #understandable dialog; #card beeping; #remote siren;
Cafe or Restaurant: small cafe or restaurant (indoor)	#low volume noise; #blurred dialog; #live music;
Car: driving or traveling as a passenger, in the city (vehicle)	#dashboard tick; #engine noise; #noise of bumpy road;
City center (outdoor)	#intense low frequency noise; #mixed noise;
Forest path (outdoor)	#slight background noise; #radio jamming sound; #remote bird chirp; #walking on the dirt road; #unclear dialog;
Grocery store: medium size grocery store (indoor)	#clear dialog; #background music; #door opening; #noise from rotating parts;
Home (indoor)	#kitchen noise; #clear dialog;
Lakeside beach (outdoor)	#breeze; #water noise; #unclear dialog;
Library (indoor)	#walking indoor; #chair noise; #mouse clicking; #unclear dialog;
Metro station (indoor)	#unclear dialog; #air releasing sound; #train coming; #train starting up; #train passing by;
Office: multiple persons, typical work day (indoor)	#mouse clicking; #typing noise; #unclear dialog;
Residential area (outdoor)	#car passing by; #bird chirping;
Train (traveling, vehicle)	#super low frequency noise; #radio jamming; #noise of wheel hitting track seams;
Tram (traveling, vehicle)	#unclear dialog; #intense vibration; #noise from rotating parts;
Urban park (outdoor)	#bird chirping;

Table1. Classes and content

There are 1170 audio clips, each 30 seconds long. We use the evaluation split from the contest and reserve 290 for testing (25%), and 880 for training. Since there are more than one audio clips from each location, we make sure no one location appears in both training and testing set. Note that this test set we use is part of the development set from the contest, not the real test set (which has not yet been released). Therefore our results might be different from the final contest result.

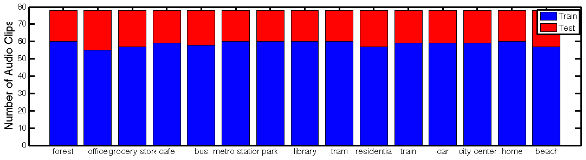


Figure 1: Class distribution of training and test data.

Within the 880 audio clip training set we also do 8-fold cross validation for model selection and parameter tuning, and again we made sure that no one location appears in both train set and validation set to have better generalization accuracy estimates.

2.2. Features

Using signal processing techniques (Section 2.1), we create 4 sets of features:

1. MFCC: We take 20 Mel-frequency cepstral coefficients over window length 0.04 second. We augment the feature with first and second order differences, resulting in a 60-dimensional vector.
2. Smile983: We use OpenSmile (Eyben et al. (2010)) to generate MFCC, Fourier transforms, zero crossing rate, energy, and pitch, among others. We also compute temporal dynamics feature, and second order dynamics features. After removing features with less information, this results in 6573 features. We select 983 features recommended by domain experts to create Smile983 feature (983 dimensional). Note this is much larger feature set than MFCC features and each feature represents longer time windows: 0.1 second.
3. Smile6k: This is the full 6573-dimensional feature set extracted from OpenSmile as described above.
4. Spectrogram: We compute spectrogram (section 2.1) and truncate at frequency 750Hz (the recording was 44100Hz).

All features are standardized to have zero mean and unit variance on the training set. The same standardization is applied at validation and test time. For each audio clip (train or test), our processing pipeline consists of the followings: 1) Break up the audio clips into windows of 4_100ms segments; 2) Apply transforms to each audio clips to extract feature representation; 3) For non-temporal model such as Gaussian Mixture Model, we treat each feature as a training example. For temporal models such as recurrent neural network, we consider a sequence of features as one training example; 4) At test time, we apply the same pipeline and as training and break the audio clip as multiple instances (feature or sequence of features), and the likelihood of a class label for a test audio clip is the sum of predicted class likelihood for each segment. The class with the highest predicted likelihood is the predicted label for the test audio clip.

2.3. Models and Hyperparameter Tuning

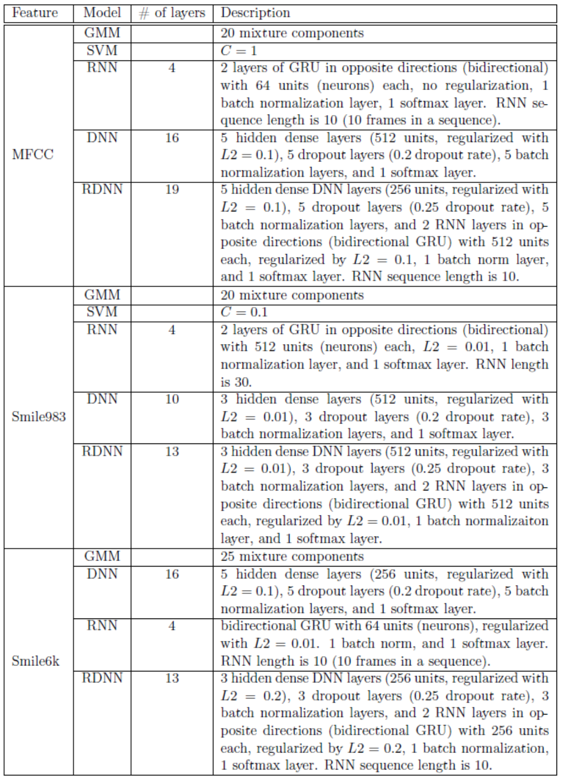
In this project, we implemented following models (table 1) with Keras library (Chollet(2015)) built on Theano (Bastien et al. (2012)), using 4 Titan X GPU on a 32GB memory, Intel Core i7 node.

2.4. Ensemble Methods

An ensemble is a collection of models whose predictions are combined by different mechanism. An ensemble of classifiers helps to cancel out inaccurate predictions, thus, tends to be more accurate than any of its individual members. There are various ensemble methods to generate more accuracy sets of models. In this project, we apply six different ensemble methods to combine predictions. Also, we apply hierarchical ensemble methods to further stabilize our predictions. These six ensemble methods are random forest Algorithm, Extremely Randomized Trees, Adaboost, Gradient Tree Boosting, weighted average probabilities (soft voting with hand craft weight) and model selection method (Caruna (2004)).

Here we ensemble deep learning models mentioned above. In total, we have twenty models for the problem, five different architectures, with four folds on the dataset. Most models have good performance (better than the baseline), some of then have mediocre and poor result. In theory, these ensemble methods could directly pick reasonable combination of models. However, we still divide these models into two sets, the elite model set and the mediocre model set. Also, to further stabilize the result of outcome, we also ensemble different ensemble results.

Base on these ensemble methods, we generate four submissions. Submission1 is the weighted average ensemble on four model selection ensemble methods. Submission2 is the weighted average ensemble on all twenty ensemble methods. Submission3 is the weighted average ensemble on numerous models trained on the whole dataset. Submission4 is the single best performance model.



Feature	Model	# of layers	Description
MFCC	GMM		20 mixture components
	SVM		$C = 1$
	RNN	4	2 layers of GRU in opposite directions (bidirectional) with 64 units (neurons) each, no regularization, 1 batch normalization layer, 1 softmax layer. RNN sequence length is 10 (10 frames in a sequence).
	DNN	16	5 hidden dense layers (512 units, regularized with $L2 = 0.1$), 5 dropout layers (0.2 dropout rate), 5 batch normalization layers, and 1 softmax layer.
	RDNN	19	5 hidden dense DNN layers (256 units, regularized with $L2 = 0.1$), 5 dropout layers (0.25 dropout rate), 5 batch normalization layers, and 2 RNN layers in opposite directions (bidirectional GRU) with 512 units each, regularized by $L2 = 0.1$, 1 batch norm layer, and 1 softmax layer. RNN sequence length is 10.
Smile983	GMM		20 mixture components
	SVM		$C = 0.1$
	RNN	4	2 layers of GRU in opposite directions (bidirectional) with 512 units (neurons) each, $L2 = 0.01$, 1 batch normalization layer, and 1 softmax layer. RNN length is 30.
	DNN	10	3 hidden dense layers (512 units, regularized with $L2 = 0.01$), 3 dropout layers (0.2 dropout rate), 3 batch normalization layers, and 1 softmax layer.
	RDNN	13	3 hidden dense DNN layers (512 units, regularized with $L2 = 0.01$), 3 dropout layers (0.25 dropout rate), 3 batch normalization layers, and 2 RNN layers in opposite directions (bidirectional GRU) with 512 units each, regularized by $L2 = 0.01$, 1 batch normalizaiton layer, and 1 softmax layer.
Smile6k	GMM		25 mixture components
	DNN	16	5 hidden dense layers (256 units, regularized with $L2 = 0.1$), 5 dropout layers (0.2 dropout rate), 5 batch normalization layers, and 1 softmax layer.
	RNN	4	bidirectional GRU with 64 units (neurons), regularized with $L2 = 0.01$. 1 batch norm, and 1 softmax layer. RNN length is 10 (10 frames in a sequence).
	RDNN	13	3 hidden dense DNN layers (256 units, regularized with $L2 = 0.2$), 3 dropout layers (0.25 dropout rate), 3 batch normalization layers, and 2 RNN layers in opposite directions (bidirectional GRU) with 256 units each, regularized by $L2 = 0.2$, 1 batch normalization, 1 softmax layer. RNN sequence length is 10.

Table 1: Models selected from cross validation for each model-feature combination.

3. RESULTS

Figure 4 shows the test accuracy for 5 classifiers over 3 features1. The model parameters are selected via cross validation and is detailed in Table 1. We perform 10000 bootstraps on the test set to estimate the standard deviation, and the difference between the test errors are statistically significant within each feature group using T-test. We point out that GMM with MFCC feature is the official baseline provided in IEEE challenge on Detection and Classification of Acoustic Scenes and Events (DCASE), which achieves a mean accuracy of 67:6%, while our best performing model (hierarchical DNN with Smile6k) achieves mean accuracy of 82.3%.

Figure 5 are the confusion matrixes of the test result between 15 classes based on DNN on Smile6k feature, which is the best performing setting from Figure 4.

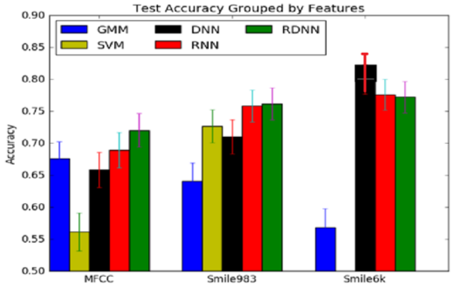


Figure 4: Test accuracy of GMM, SVM, RNN, DNN(including hierarchical model), and RDNN on three features: MFCC, Smile983, and Smile6k feature. The model parameter details are in Table 1. Note that for Smile6k feature Liblinear SVM could not finish computation in reasonable amount of time and thus not included.

4. DISCUSSION

Figure 4 shows that feature representation is critical for classifier performance. For each neural network model (RNN, DNN, RDNN) larger set of features extracted from signal processing pipeline improves performance. Among the neural network models, it is interesting to note that temporal models (RNN and RDNN) outperforms DNN using MFCC and Smile983 features, but DNN outperforms RNN and DNN on Smile6k features and achieves the best accuracy among all models. It is possible that with limited feature representation (e.g., MFCC and Smile983), modeling temporally adjacent pieces enhances local feature representation and thus improves performance in those cases. However, with sufficiently expressive feature (e.g., Smile6k), the temporal modeling become less important, and it becomes more effective to model local dynamics rather than long-range dependencies.

This observation is somewhat surprising as we originally expected temporal models to outperform static model (DNN) because sound is time-series data. A more careful consideration reveals that, unlike speech, which has long range dependency (a sentence utterance could span 6_20 seconds), environment sounds generally lacks a coherent context, as events in environment occurs more or less randomly from the listener's perspective. To put it another way, a human listener of environmental noise is unlikely able to predict what sound will occur next in an environment, in contrast to speech. (Even though there are speeches and chatters in environmental sounds, it is the presence of speech rather than the content of speech that's instrumental for our task.) This weak global dependency property in a time series data is not unique to this problem setting. Kim et al. (2015) made a similar observation in the context of facial expression synthesis based on speech. They found that even though facial motion is temporal, it is more beneficial to simply model the local dynamics with decision tree, which outperforms HMM, LSTM, and other temporal models. Another example is edge detection in the image. While different parts of image could be related to each other, Dollár and Zitnick (2013) shows that it is more beneficial to model local patch dynamics than to consider the picture as a whole in performing edge detection.

Regarding to the non-neural network models, the performance of GMM decreases with increasing dimension, which is expected due to "curse of dimensionality". That is, in high dimensional space, the volume grows exponentially while the number of available data stays constant, leading to highly sparse sample. SVM's performance is very poor for MFCC, as linear SVM has limited model capacity. With increasing feature dimension (Smile983) SVM performance improves.

Finally, the confusion matrixes in Figure 5 shows that most locations are relatively easy to identify, such as beach, bus, and car. However, some locations are fairly difficult to distinguish, such as park and residential area, or home and library. Also, the model accuracy is highly dependent on how we divide the folds of the dataset. But at least, these are consistent with our intuition, as the less distinguishable locations in fact sound like each other (parks could be close to residential area; both home and library could be rather quiet). This is an evidence that the classifiers we train indeed learn some characteristics of the environment sounds.

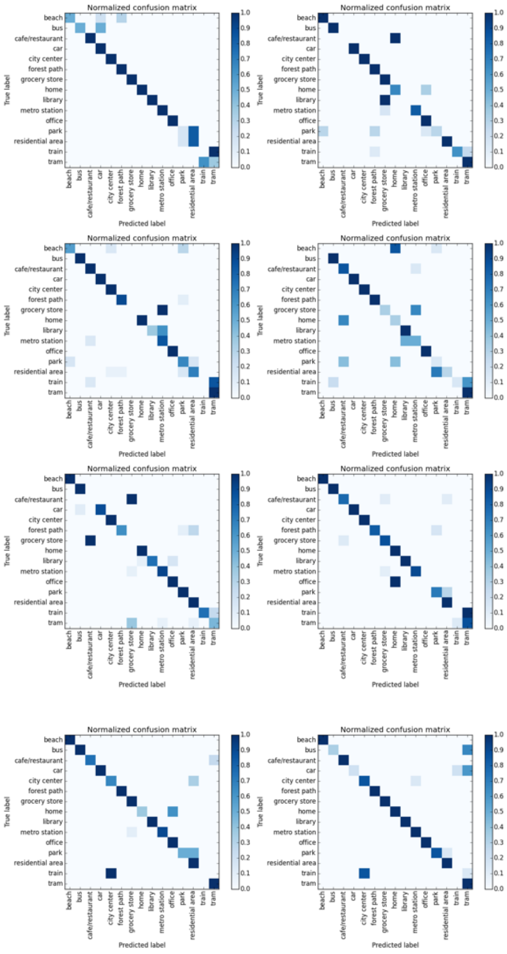


Figure 5: Confusion Matrix of DNN using Smile6k feature

5. FUTURE WORK

Since our results could be affected due to the limited data, data augmentation is expected to be very helpful. However, data augmentation in the context of environment sound recognition is trickier than in speech recognition and image classification, because noise is often part of the environment sound, and simply adding noise could change the characteristics of the environment. One possible way to avoid that is to warp time without changing speed using phase vocoder. Another possibility to enhance data is to use other environmental sound data to perform joint training on the two datasets. For example we can let two tasks shares the same feature extraction and DNN pipeline, but use separate classifiers or softmax layers at the end for each task. While we can't use external data for the purpose of DCASE challenge, but it'd be an interesting direction to improve performance with limited (labeled) data outside of the competition.

6. CONCLUSION

In this work, we apply 5 models to acoustic scene recognition: Gaussian mixture model (GMM), Support Vector Machine (SVM), Deep Neural Network (DNN), Recurrent Neural Network (RNN), Recurrent Deep Neural Network (RDNNWe use 4 sets of features extracted using signal processing techniques: MFCC (60-dim), Smile983 (983-dim), Smile6k (6573-dim), and spectrogram.

We find that deep learning models compare favorably to traditional models (GMM and SVM). Specifically, GMM with MFCC feature, the baseline model provided by DCASE contest, achieves 67.6% test accuracy, while the best performing model (hierarchical DNN with Smile6k feature) reaches 82.3% test accuracy. RNN and RDNN generally have performance in the range of 68_77%, while SVM varies between 56_73%. No single model outperforms all other models across all feature sets, showing that model performance varies significantly with feature representation. The fact that the best performing model is the non-temporal DNN model is evidence that environmental (scene) sounds don't necessarily exhibit strong temporal dynamics. This is consistent with our day-to-day experience that environmental sounds tend to be random an unpredictable.

7. REFERENCES

[1] Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

[2] Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I., Bergeron, A., Bouchard, N., Warde-Farley, D., and Bengio, Y. (2012). Theano: new features and speed improvements. arXiv preprint arXiv:1211.5590.

[3] Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. Neural Networks, IEEE Transactions on, 5(2):157–166.

[4] Bishop, C. M. (2006). Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[5] Cakir, Emre, et al. "Polyphonic sound event detection using multi label deep neural networks." 2015 international joint conference on neural networks (IJCNN). IEEE, 2015.

[6] Mesaros, Annamaria, et al. "Sound event detection in real life recordings using coupled matrix factorization of spectral representations and class activity annotations." 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015.

[7] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H.,and Bengio, Y. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

[8] Chollet, F. (2015). keras. https://github.com/fchollet/keras.

[9] Ciresan, D., Meier, U., and Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. In Computer Vision and Pattern Recognition (CVPR), 2012, pages 3642–3649. IEEE.

[10] John Duchi, Elad Hazan and Yoram Singer. (2011). Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. Journal of Machine Learning Research 12 (2011) 2121-2159

[11] Clevert, D.-A., Unterthiner, T., and Hochreiter, S. (2015). Fast and accurate deep network learning by exponential linear units (elus). arXiv preprint arXiv:1511.07289.

[12] Dollár, P. and Zitnick, C. (2013). Structured forests for fast edge detection. In Proceedings of the IEEE International Conference on Computer Vision, pages 1841–1848.

[13] Eyben, F., Wöllmer, M., and Schuller, B. (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.

[14] Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. (2008). Liblinear: A library for large linear classification. J. Mach. Learn. Res., 9:1871–1874.

[15] Gales, M. and Young, S. (2008). The application of hidden markov models in speech recognition. Foundations and trends in signal processing, 1(3):195–304.

[16] Graves, A., Mohamed, A.-r., and Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, pages 6645–6649. IEEE.

[17] Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., et al. (2014). Deep speech: Scaling up end-to-end speech recognition. arXiv preprint arXiv:1412.5567.

[18] He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE International Conference on Computer Vision, pages 1026–1034.

[19] Hinton, G. (2012). Neural networks for machine learning. http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf.

[20] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8):1735–1780.

[21] Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.

[22] Kim, T., Yue, Y., Taylor, S., and Matthews, I. (2015). A decision tree framework for spatiotemporal sequence prediction. In Proceedings of the 21th ACM SIGKDD International

Conference on Knowledge Discovery and Data Mining, pages 577–586. ACM.

[23] Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[24] Mesaros, A. (2016). 2016 dcase challenge. http://www.cs.tut.fi/sgn/arg/dcase2016/.

[25] Mikolov, T., Kombrink, S., Deoras, A., Burget, L., and Cernocky, J. (2011). Rnnlmrecurrent neural network language modeling toolkit. In Proc. of the 2011 ASRU Workshop, pages 196–201.

[26] Pinheiro, P. H. and Collobert, R. (2013). Recurrent convolutional neural networks for scene parsing. arXiv preprint arXiv:1306.2795.

[27] Ranft, R. (2004). Natural sound archives: past, present and future. Anais da Academia Brasileira de Ciências, 76(2):456–460.

[28] Reynolds, D. A., Quatieri, T. F., and Dunn, R. B. (2000). Speaker verification using adapted gaussian mixture models. Digital signal processing, 10(1):19–41.

[29] Roma, G., Nogueira, W., Herrera, P., and de Boronat, R. (2013). Recurrence quantification analysis features for auditory scene classification. IEEE AASP Challenge: Detection and Classification of Acoustic Scenes and Events, Tech. Rep.

[30] Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[31] Sutskever, I., Martens, J., Dahl, G., and Hinton, G. (2013). On the importance of initialization and momentum in deep learning. In Proceedings of the 30th international conference on machine learning (ICML-13), pages 1139–1147.

[32] Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1701–1708.

[33] Van den Oord, A., Dieleman, S., and Schrauwen, B. (2013). Deep content-based music recommendation. In Advances in Neural Information Processing Systems, pages 2643–2651.

[34] Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3156–3164.

[35] Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X., Moore, G., Odell, J., Ollason, D., Povey, D., et al. (1997). The HTK book, volume 2. Entropic Cambridge Research Laboratory Cambridge.

[36] Zeiler, M. D., Ranzato, M., Monga, R., Mao, M., Yang, K., Le, Q. V., Nguyen, P., Senior, A., Vanhoucke, V., Dean, J., et al. (2013). On rectified linear units for speech processing. In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, pages 3517–3521. IEEE.

[37] Caruana, Rich, et al. "Ensemble selection from libraries of models."Proceedings of the twenty-first international conference on Machine learning. ACM, 2004.