INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

An Adaptive Multi-Band System for Low Power Voice Command Recognition

Qing He[1], *Gregory W. Wornell*[1], *Wei Ma*[2]

[1]EECS & RLE, MIT, Cambridge, MA 02139, USA
[2]Texas Instruments, Santa Clara, CA, 95051, USA
qinghe@mit.edu, gww@mit.edu, wei.ma@ti.com

Abstract

A complete voice-driven experience in applications such as wearable electronics requires always-on keyword monitoring, which is prohibitively power consuming using current speech recognition methods. In this work, we propose an ultra-low power voice command recognition system that is designed to recognize short commands such as 'Hi Galaxy'. To achieve power-efficient designs, the system uses adaptive feature pre-selection such that only a subset of all available features are selected and extracted based on the noise spectrum. The back-end classifier, supporting adaptive feature selection, is enabled by a novel multi-band deep neural networks (DNNs) model that processes only the selected features at each decision. In experiments, our adaptive scheme achieves comparable accuracy and improved efficiency using an average of 5 spectral feature bands, than a generic fully-connected DNNs model using the full speech spectrum. The system makes a recognition decision every 40ms on 1.2s of buffered speech and consumes $\sim$ 230μW of power, thus promising low-power, low-complexity and robust application-specific voice recognition.

Index Terms: Deep neural networks, keyword spotting, low power, band selection

1. Introduction

The complexity of current speech recognition algorithms exceeds the power constraints and computation capability of typical mobile devices such as smart phones and watches. While primary processing can be relegated to powerful hosts that reside in the cloud, system activation remains problematic for a completely voice-driven experience. Hence, there are rising interests in finding simple, low-power solutions for the task of voice wake-up.

Prior work relevant to voice wake-up is found in the keyword spotting (KWS) literature [1, 2, 3, 4]. In KWS, the processing of speech waveforms into keyword decisions follows a two-stage pipeline: (1) a feature extraction front-end transforms raw speech into low-dimensional features, and (2) a back-end classifier decides from features whether a candidate keyword was uttered.

For the first stage, Mel-frequency cepstral coefficients (MFCCs) are widely used as features [5]. When used as input into the DNNs, raw spectral features (i.e., filterbank features) are found to yield better performance than further transformed cepstral coefficients [6, 7]. The second stage involves classical pattern recognition to distinguish candidate classes. Template-based algorithms match features from candidate class samples directly to query features [4, 8], while model-based algorithms render the speech features as statistical emissions from a class [9, 10, 11]. Recent developments toward using DNNs for KWS suggest that DNN models significantly improve accuracy over conventional HMM models [1].

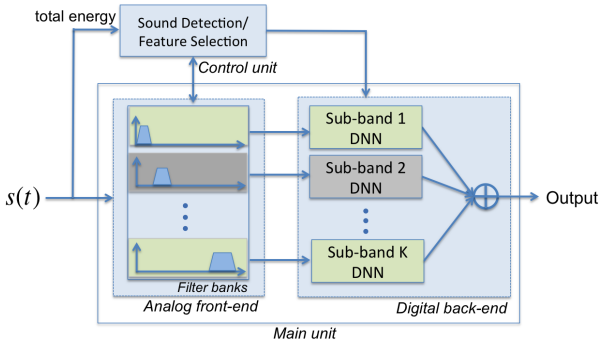


Figure 1: *System block diagram: the filter-bank outputs from each sub-band is fed to its corresponding sub-module of the classifier. The final recognition decision is a weighted sum of sub-band decisions. During training all the sub-bands are turned on. During recognition, the control unit pre-selects a subset of all available sub-bands and only the selected sub-band features (e.g., the green blocks) are extracted and processed.*

While these existing KWS systems achieve excellent accuracy, they are prohibitively power-consuming for standalone devices due to the extraction and processing of high-dimensional features (e.g., 40-dim MFCC at 100 frames per second). In this paper, we propose an adaptive voice command recognition system that achieves ultra-low power consumption, accuracy, and robustness by enabling the dynamic processing of an adaptively selected set of features. Specifically, an analog front-end (AFE) extracts spectral features from only the desirable sub-bands, which are selected based on the background noise, and a novel multi-band DNN classifier completes the recognition task with only the selected sub-band features as partial input. By adjusting the features in use judiciously, the average system computation complexity is significantly reduced without sacrificing recognition accuracy for the task of voice-command recognition. By selecting features based on signal quality and actively turning off noisy bands, the system achieves better noise robustness than using all available spectral features.

We describe the overall system in Sec. 2 and introduce the multi-band DNN model in Sec. 3. Sec. 4 describes experiments and practical measurements. Assume continuous triggering, our system continuously listens to and adapts to the surrounding environment in order to recognize a wake-up command such as 'Hi, Galaxy'. A command-level decision on a buffer of 1.2s of sound information is made every 40ms using $\sim$ 230μW of power. The recognition accuracy is 99% under quiet conditions, and is approximately 97% under various background noise.

http://dx.doi.org/10.21437/Interspeech.2016-1562

2. System description

Fig. 1 shows the high-level system architecture. The main unit consists of the feature extraction AFE and the multi-band DNN back-end. An external control unit wakes up the main unit whenever an incoming signal of sufficient total power is detected, and more importantly, it adaptively selects a small number (e.g., $\sim$ 5) of spectral features based on criteria such as the in-band signal-to-noise ratio (SNR) of the sub-bands.

Both the AFE and the back-end are designed to efficiently support adaptive band-selection so that only the selected features are extracted and processed to make the recognition decision. The AFE consists of a band-pass filter-bank, which extracts the contents within the selected frequency sub-bands of raw speech. The AFE outputs are the accumulated power within each time frame for the selected sub-bands. The backend DNN-based classifier employs a multi-band model. In contrast to the conventional approach of interpreting a time-sequence of spectral feature vectors as a single super-vector, the multi-band model performs classification disjointly for each sub-band using their corresponding time-sequence of single-band features (as shown in Fig. 1). The final output is a weighted sum of the sub-band decisions. The multi-band model not only enables adaptive feature selection, but the sparsely-connected structure also requires less computation than the fully-connected DNN (e.g., [1]) given a fixed feature dimension.

System support for adaptive processing is highly beneficial because speech content for recognition is redundant in the spectral sub-bands, and a subset of all available bands can be sufficient for the task of voice wake-up. Hence, we can scale processing power by using fewer sub-band features when there is no background noise and more features when there is noise. In addition, adaptive feature selection can mitigate the loss of granular SNR as in the conventional approach, which concatenates all sub-band features into a super-vector regardless of noise conditions, resulting in poor recognition even when only a single band may be corrupted. In contrast, adaptive selection results in better robustness by actively discarding the noisy features and retaining only the high quality ones.

3. Adaptive multi-band DNN

Deep learning with neural networks has demonstrated state-of-the-art performance in a range of speech recognition tasks [12, 1]. In contrast to the conventional approach of modeling the time-frequency features as a whole using one fully-connected DNNs, we use a multi-band DNN.

3.1. The multi-band DNN

As illustrated in Fig. 2, the time-frequency features are divided into separate sub-bands $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$. Each of $\mathbf{x}_i$ represents a time-sequence of filter-bank features within a single sub-band over the duration of a keyword (e.g., $\sim$ 1s). Each sub-band is then modeled with a fully-connected DNN whose top layer has two nodes, representing the 'keyword' and 'out-of-vocabulary' (OOV) classes. The top layers of all sub-band DNNs are then connected to the final decision output layer.

The multi-band DNNs model offers the following key benefits. *Adaptive band-selection:* When the sub-band parameters are trained disjointly, the multi-band model can be used to support adaptive band-selection such that only the selected sub-bands are active. *Model size:* Let N denote the total number of frequency bands. Given a fixed number of hidden layers and a fixed number of nodes per layer, the number of edges in the multi-band DNN model increases linearly with N, whereas it increases with N^2 in the fully-connected DNNs because nodes corresponding to different bands are cross-connected. As a result, the multi-band DNN model requires a factor of N fewer multiplications for the recognition task given fixed feature and hidden-layer dimensions; and its sparse structure requires less data for model training. Combing adaptive band-selection and the multi-band model, system complexity and power consumption of back-end recognition are reduced.

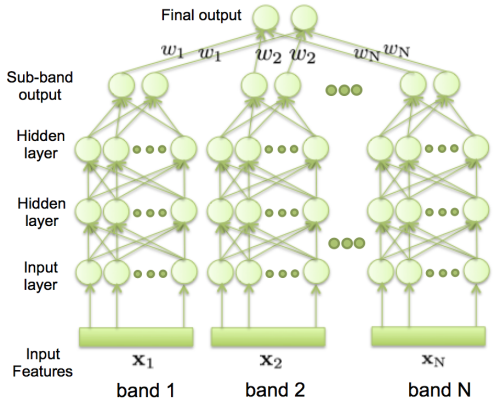


Figure 2: *The multi-band DNNs model: each sub-band is modeled with a separate fully-connected DNN and the individual sub-band decisions are merged at the sub-band output layer.*

3.2. Training and classification

Training: The parameters for each sub-band DNN can be trained in a substantially disjoint fashion. We take two approaches for training. In the first approach, each sub-band DNN is treated as an independent classifier trained with the back-propagation algorithm, followed by the weighted-majority algorithm [13] to obtain the weights of output layer with all sub-bands simultaneously presented. Higher weights are assigned to sub-bands with better accuracy. In the second approach, the sub-band DNNs are first trained independently. Then, the parameters of the individual sub-bands are fine-tuned in sequence using the back-propagation algorithm along with AdaBoost[14, 15, 16], which combines a set of weak classifiers to construct a strong classifier. At each iteration, the weights of the training samples are updated based on the errors made by the current sub-band classifier, and these weights are used to adjust the back-propagating error for each sample when training the next classifier. At the end, sub-band weights at the top layer are obtained as a result from AdaBoost.

Classification: As illustrated in Fig. 2, the sub-band decisions are combined as a weighted sum at the final output layer. Let $S \subset \{1, ..., N\}$ denote the set of active bands, $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ denote inputs to the N sub-bands, $\{w_1, \ldots, w_N\}$ denote the weights at the sub-band outputs, and $Y = \{y_1, y_2\}$ denote labels for the two output classes. Let $h_n(\mathbf{x}_n, y_i)$ represent the soft-decision output at sub-band n. Then the final output is a weighted sum of the active sub-band decisions: $\sum_{n \in S} w_n h_n(\mathbf{x}_n, y_i)$.

3.3. SNR-based adaptive selection

In the case where sub-bands are selected based on SNR, the active band set S for classification is chosen as follows. We first estimate the sub-band in-band SNRs using the spectrum power distribution obtained from speech training samples and real-time noise power measurements in each band. Let θ_{SNR} de-

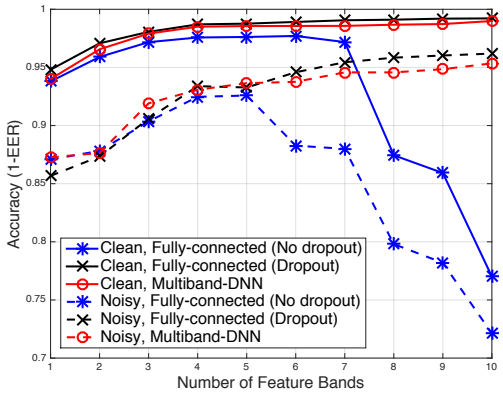


Figure 3: *Performance comparison: the multi-band DNNs offer accuracies comparable to the fully-connected DNNs. Unlike the fully-connected model, the multi-band model does not suffer from over-fitting when the number of bands increases.*

note the minimum desired in-band SNR threshold and let $K_{\max}$ denote the maximum number of bands to be turned on. The active set S includes the bands with the best in-band SNRs such that a maximum of $K_{\max}$ bands are chosen and all the bands in S must have SNR higher than θ_{SNR}. If S is empty, then use the single band that has the highest in-band SNR.

4. Experiments

4.1. Experiment setup

We analyze the multi-band DNNs model in two sets of experiments. First, we investigate how well the multi-band DNN structure can model speech commands by comparing it with the conventional fully-connected DNN, in which all the sub-bands are inter-connected. In this case, we fix the band selections and analyze the performance when the same sub-bands are used for both training and classification. The band selection is chosen from the 13-band Mel-frequency filter banks in a way that yields the best accuracy among all choices of the same subset cardinality. We analyze the performance as the number of sub-bands increases. Second, we study the system performance with adaptive feature selection (Sec. 3.3), with SNR threshold set to $\theta_{\text{SNR}} = 5$ dB and maximum band usage $K_{\max} = 5$.

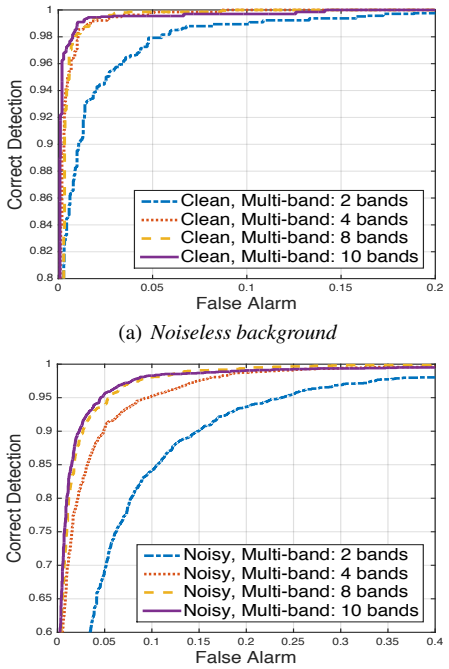
4.1.1. Data sets

The clean data set includes 3000 positive examples of the keyword 'Hi Galaxy' recorded by 100 different speakers, and 32k negative examples (12k examples of other commands and 20k short phrases taken from audio books and audio shows). The noisy-condition data sets are generated by adding to each sample of the clean data set either a recording of real noise data or a pseudo-noise sample of defined spectral statistics.

4.1.2. Model size and algorithm implementation

Recognition is performed at the command level with 1.2s of audio content. The features are extracted at a frame rate of 10ms and down-sampled to 50 samples per second. As a result, the input feature dimension is 60 for each of the K sub-bands of the multi-band model and it is $K \times 60$ for the fully-connected model (K is the number of active bands). Both the multi-band DNN and the fully-connected DNN have 3 hidden layers, whose dimensions reduce by $1/2$ at each layer.

In each simulation configuration, a random 90% of samples



(a) *Noiseless background*

(b) *Noisy background at 0dB SNR*

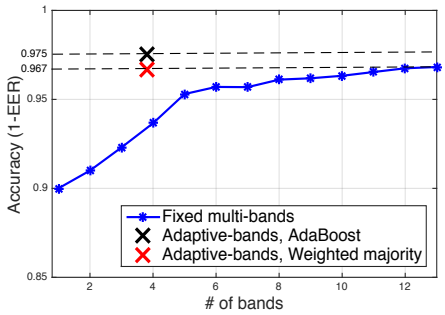
Figure 4: *ROC corresponding to operating points from Fig. 3. Recognition accuracy improves as the number of bands increases and stops at around 4 bands when there is no background noise and at 8 bands under noisy conditions.*

are used for training and the remaining 10% are withheld for testing. This is repeated 10 times and the results are averaged. The back-propagation algorithm is implemented with the mean-square-error cost function and random parameter initialization. The learning rate is 0.01 and the training procedure terminates when the gradient is less than 10^{-7} or when it exceeds 1000 iterations. For the fully-connected model, when it is trained with dropout[17], the probability of retention is 0.9 for the input layer and 0.5 for the hidden layers.

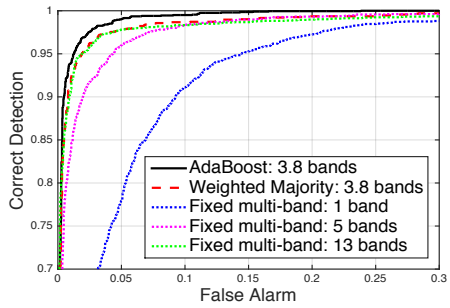
4.2. The fixed multi-band DNN experiments

In these experiments, noisy samples are generated by adding scaled real noises (e.g., babble noise, car noise, wind noise, radio and audio book noise) so that the total SNR is 0dB.

Fig. 3 plots the accuracy (1 - EER) of the fixed multi-band model and the fully-connected model as a function of the number of frequency bands under quiet and noisy conditions. There are three main points to note from Fig. 3. First, even though the multi-band model presumes a disjoint structure among different bands, the multi-band model yields similar recognition accuracy as the fully-connected model for our task of single command recognition. Secondly, the performance of the multi-band model increases steadily with the number of bands and saturates at 4 bands and 8 bands under noiseless and noisy conditions, respectively. Similar behavior can be seen from the receiver operating characteristic (ROC) curves shown in Fig. 4. This implies that, computation resources can be optimized by using fewer bands under quiet conditions and by including more bands when noise is present. Lastly, the multi-band structure allows the training data size per band to be independent of the number of bands, whereas the fully-connected model requires an increasing number of training samples with increasing number of bands N. This is illustrated in Fig. 3, where it shows

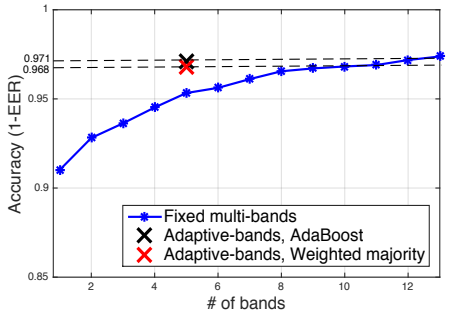


(a) *Pseudo noise: EER*

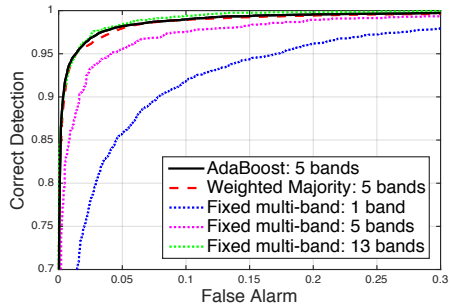


(b) *Pseudo noise: ROC*

Figure 5: *Experiments under pseudo-noise: with* < 4 *bands on average, the adaptive multi-band method achieves an accuracy of* 97.5%*, outperforming the* 13 *band fixed multi-band method.*



(a) *Real noise: EER*



(b) *Real noise: ROC*

Figure 6: *Experiments under real noise: with 5 frequency bands, the adaptive multi-band approach achieves comparable performance as the* 13 *band fixed multi-band method.*

that, when the fully-connected DNN model is trained without dropout, over-fitting occurs when the number of bands exceeds a certain threshold.

4.3. The adaptive multi-band DNN experiments

Here, two types of noisy samples are used. First, pseudo-noise are added to clean samples. The spectrum of the noise samples are shaped to be band-wise white in 500Hz bands in the range of 0-8kHz with in-band SNR randomly chosen between -10dB and 15dB. In the second set, real noise (wind and car noise) are added to clean samples such that the total SNR is randomly chosen between -5dB and 10dB.

Fig. 5(a) shows the performance of two adaptive multi-band schemes relative to the fixed multi-band scheme under pseudo-noise. On average, fewer than 4 frequency bands are chosen by adaptivity. The AdaBoost method and the weighted-majority method yield an EER accuracy of 97.5% and 96.7%, respectively. The best performance for the fixed multi-band method is achieved with 13 bands, and yields an accuracy of 96.8%, which is slightly less than the AdaBoost method, demonstrating the substantial benefits of rejecting noisy bands. Similar observations are shown in the ROC plot in Fig. 5(b).

Fig. 6(a) plots the results under wind and car noises, which is common for our application and has the special characteristics of concentrating narrowly in low frequencies. Similar to the case with pseudo-noise, by adaptively selecting a subset of 5 features using the procedure described in Sec. 3.3 and Sec. 4.1, the adaptive system achieves comparable performance as the fixed (non-adaptive) approach, which uses more than twice the number of features.

4.4. Power evaluation on hardware

The total system power consumption is evaluated as the sum of the front- and back-ends power consumption. The front-end is designed by Texas Instruments. It has a fixed power consumption of 150μW going to the total energy thresholding unit and the 13-band analog filter-bank and an additional power cost of 10μW per active band for feature extraction. The digital back-end is implemented on a Cortex-M0 processor. The firmware implementation for the algorithm and data consumes less than 40kB memory and under 10μW of power per band. The total power consumption increases linearly with the number of active sub-bands. For example, with adaptive band-selection using an average of ~ 4 bands, the entire system would consume $\sim 230\mu$W when continuously triggered.

5. Conclusion and future work

In this paper, we presented a low power voice-command recognition system equipped with adaptive feature selection and multi-band DNN classification. Without degrading the recognition accuracy, the system offers simpler processing, improved noise robustness and lower power consumption compared to the conventional approach of using fixed features with fully-connected DNNs. As a next step, more sophisticated in-band SNR thresholding and band-selection procedures can be developed based on joint analysis of the speech spectrum and noise properties in order to further improve the system robustness under general noise.

6. Acknowledgment

The authors would like to thank Kilby Labs at Texas Instruments for supporting this research work. This work was supported in part by Texas Instruments, by NSF under Grant No. CCF-1319828, and by Systems on Nanoscale Information fabriCs (SONIC), an SRC STARnet Center sponsored by MARCO and by DARPA.

7. References

[1] G. Chen, C. Parada, and G. Heigold, "Small-footprint keyword spotting using deep neural networks," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Florence, IT, May 2014.

[2] J. G. Wilpon, L. Rabiner, C.-H. Lee, and E. Goldman, "Automatic recognition of keywords in unconstrained speech using hidden Markov models," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 38, no. 11, pp. 1870–1878, 1990.

[3] I. Szöke, P. Schwarz, P. Matějka, L. Burget, M. Karafiát, and J. Černockỳ, "Phoneme based acoustics keyword spotting in informal continuous speech," in *Proc. Text, Speech and Dialogue*, Karlovy Vary, CZ, September 2005.

[4] Y. Zhang and J. R. Glass, "Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams," in *Proc. IEEE Workshop on Automatic Speech Recognition & Understanding*, Merano, IT, November 2009.

[5] B. Gold, N. Morgan, and D. Ellis, *Speech and audio signal processing: processing and perception of speech and music*. John Wiley & Sons, 2011.

[6] Q. He, G. Wornell, and W. Ma, "A low-power text-dependent speaker verification system with narrow-band feature preselection and weighted dynamic time warping," in *Proc. Odyssey 2016: The Speaker and Language Recognition Workshop*, Bilbao, Spain, June 2016, pp. 1–8.

[7] L. Deng, J. Li, J.-T. Huang, K. Yao, D. Yu, F. Seide, M. Seltzer, G. Zweig, X. He, J. Williams *et al.*, "Recent advances in deep learning for speech research at Microsoft," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Vancouver, Canada, May 2013.

[8] A. S. Park and J. R. Glass, "Unsupervised pattern discovery in speech," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 186–197, 2008.

[9] R. C. Rose and D. B. Paul, "A hidden Markov model based keyword recognition system," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Albuquerque, US, April 1990.

[10] J. R. Rohlicek, W. Russell, S. Roukos, and H. Gish, "Continuous hidden Markov modeling for speaker-independent word spotting," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Glasgow, UK, May 1989.

[11] M.-C. Silaghi, "Spotting subsequences matching an HMM using the average observation probability criteria with application to keyword spotting." in *Proc. National Conf. Artificial Intelligence*, vol. 20, no. 3, Pittsburgh, US, July 2005, p. 1118.

[12] A.-r. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 14–22, 2012.

[13] N. Littlestone and M. K. Warmuth, "The weighted majority algorithm," *Information and computation*, vol. 108, no. 2, pp. 212–261, 1994.

[14] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *Proc. Computational learning theory*, Barcelona, Spain, March 1995.

[15] Y. Freund, R. Schapire, and N. Abe, "A short introduction to boosting," *Journal-Japanese Society For Artificial Intelligence*, vol. 14, no. 771-780, p. 1612, 1999.

[16] H. Schwenk and Y. Bengio, "Training methods for adaptive boosting of neural networks for character recognition," *Advances in Neural Information Processing Systems*, vol. 10, pp. 647–653, 1998.

[17] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.