2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW) | 979-8-3503-0261-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSPW59220.2023.10193337

LOW BIT NEURAL NETWORK QUANTIZATION FOR SPEAKER VERIFICATION

Haoyu Wang, Bei Liu, Yifei Wu, Zhengyang Chen, Yanmin Qian[†]

MoE Key Lab of Artificial Intelligence, AI Institute
X-LANCE Lab, Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, China
{fayuge, beiliu, yifei.wu, zhengyang.chen, yanminqian}@sjtu.edu.cn

ABSTRACT

With the continuous development of deep neural networks (DNN) in recent years, the performance of speaker verification systems has been significantly improved with the application of Deeper ResNet architectures. However, these deeper models occupy more storage space in application. In this paper, we adopt **A**lternate **D**irection **M**ethods of **M**ultipliers (ADMM) to realize low-bit quantization on the original ResNets. Our goal is to explore the maximal quantization compression without evident degradation in model performance. We implement different uniform quantization for each convolution layer to achieve mixed precision quantization of the entire model. Moreover, the impact of batch normalization layers in ADMM training and layer sensibility to quantization are explored. In our experiments, the 8 bit quantized ResNet152 achieved comparable results to the full-precision one on Voxceleb 1, with only 45% of original model size. Besides, we find that shallow convolution layers are more sensitive to quantization. In addition, experimental results indicate that the model performance will be severely degraded if batch normalization layers are integrated into the convolution layer before the quantization training starts.

***Index Terms*—** speaker verification, neural network quantization, model compression, mixed precision quantization

1. INTRODUCTION

In recent years, speaker verification systems using deep neural networks (DNN) as feature extractors have shown excellent performance [1, 2, 3, 4, 5]. The most typical and widely used speaker feature extraction network structures are ResNet [6] and ECAPA-TDNN [4], and their performance is improved as the network deepens. Large models have more competitive representation capabilities. Nevertheless, the application scenario of deeper architectures is limited due to oversized memory occupation. Therefore, reducing the size of deep neural networks has become a crucial research topic.

Previous studies [7, 8] have proved that parameter redundancy in convolution neural networks exists and reasonable compression of deep neural networks is theoretically feasible. Previous research has shown that knowledge distillation [9, 10, 11], model pruning [12], and model quantization [13, 14, 15, 16, 17] are all effective ways to reduce the size of deep neural networks. Among these approaches, knowledge distillation replaces large models with small-footprint models, while model pruning removes some of the model's parameters. Unlike them, quantization compression retains the structural integrity of the original model. Earlier work in the speech domain [13, 15] confirms that proper model compression has a negligible impact on model performance. These works inspired us to design a quantized embedding extractor.

In order to implement a competitive speaker verification system with a compact model, this paper explores the lowest compression ratio that can be achieved without performance degradation. The experimental results show that extreme quantization compression, such as binary quantization [17], has a more significant impact on the model accuracy. We adopt a lower compression ratio progressively to ensure the performance of the model. Meanwhile, the effect of higher compression ratios on performance is observed.

The main contributions of this paper are as follows: First, we design a feature extraction network quantization method for speaker verification system via alternate direction methods of multipliers (ADMM). The experimental results show that after 8 bit quantization compression, the quantized ResNet models can still maintain the accuracy comparable to the full-precision version with only 6% of performance degradation. Second, this paper explores the performance of mixed precision quantization models. We implement two mixed precision quantization models of different sizes and analyze their performance under high compression ratio conditions. The experimental results demonstrate that mixed precision quantization is ineffective in improving model performance compared to uniform quantization. Meanwhile, the function of batch normalization (BN) layer in model quantization is analyzed. Our results show that the BN layer is critical in the training session and cannot be removed before quantization. Finally, we find that shallow convolution layers are more sensitive to quantization.

2. RELATED WORKS

In this section, we briefly introduce the recent research results of model compression methods in speaker verification task and ADMM-based quantization approaches.

2.1. Model Compression in Speaker Verification

Some attempts at model compression have been implemented in speaker verification task. In [10, 11], teacher models can migrate their performance advantages to smaller models through knowledge distillation. Binary quantization [17] achieves the extreme compression ratio. However, the performance of binary model is significantly degraded. In previous research on speech domain, low-bit quantization is considered to enable model compression without compromising performance [13, 15, 18]. In this paper, we aim to obtain a set of speaker verification compression models with little degradation in performance through ADMM quantization.

[†]corresponding author

This work was supported in part by China NSFC projects under Grants 62122050 and 62071288, and in part by Shanghai Municipal Science and Technology Major Project under Grant 2021SHZDZX0102.

2.2. Quantization via ADMM

Alternate direction methods of multipliers (ADMM) [19] has shown its strong ability in model quantization. ADMM-based quantization allows iteratively solving the quantized parameters. Unlike previous methods of training the network from scratch [17], the ADMM algorithm can be directly applied to the pre-trained model and converge in few epochs, reflecting its excellent efficiency. [16] realizes ADMM on image recognition, and the proposed algorithm outperforms other model compression methods. [15] implements ADMM in speech recognition and achieves a model size compression ratio of up to 31 times over the baseline model.

Mixed precision quantization is a method of quantizing different parts of a model with different bit precision. Mixed precision quantization via ADMM first measures the quantization sensitivity of each module in the model [20]. It then determines the degree of compression required by each module according to the difference in sensitivity. Mixed precision quantization achieved impressive performance in speech recognition [21] and speech separation [18].

3. PROPOSED METHODS

In this section, we introduce model quantization method: ADMM for quantization model training, the design of mixed precision quantization model, and batch normalization layer processing in the procedure of model quantization.

3.1. Model Quantization

Model quantization aims to replace the original full-precision model parameters with some low-precision parameters that take up less space so that the entire model takes up less space. For n-bit uniform quantization, the set of integers N that can be selected by the quantization matrix in the model is defined as:

$$N \in \left\{0, \pm 1, \pm 2, \ldots, \pm 2^{n-1}\right\} \tag{1}$$

The quantization table is obtained by multiplying the scaling factor α and n. Under n-bit uniform quantization, the quantized parameter range in the quantization model is:

$$q = \alpha N \in \{0, \pm\alpha, \ldots, \pm\alpha \cdot (2^n)\} \tag{2}$$

For any deep neural network, we can construct a quantization table for each of its convolution layers. For example, the l-th convolution layer of the model, the value range of the quantized parameter $Q^{(l)}$ of this layer is shown as follows:

$$\boldsymbol{Q}^{(l)} = \alpha^{(l)} \boldsymbol{N}^{(l)} \in \left\{0, \pm\alpha^{(l)}, \ldots, \pm\alpha^{(l)}\left(2^{n_l - 1}\right)\right\} \tag{3}$$

where $\alpha^{(l)}$ represents the scaling factor of l-th layer, it flexibly scales the parameter coverage of the current layer after quantization according to the original parameters, n_l is the quantization precision number of l-th layer. In the case of uniform quantization, all n_l take the same value; in the case of mixed quantization, n_l varies according to the parameter sensibility of l-th layer. The quantization operation is defined as:

$$f\left(\boldsymbol{W}^{(l)}\right) = \arg\min_{Q^{(l)}} \left|\boldsymbol{W}^{(l)} - \boldsymbol{Q}^{(l)}\right| \tag{4}$$

where f denotes the quantization operation. In the training session of the quantization model, the quantization parameter Q^l of the l-th layer should be as close as possible to the parameter W of the full-precision model of the l-th layer. In our work, we only implement quantization for convolution weights.

3.2. Quantization via ADMM

Alternating Direction Multiplier Method (ADMM) [19] is a powerful optimization technique that decomposes the original optimization problem into several relatively easy-to-solve sub-optimization problems for iterative solutions. In our quantization task, to update the network parameters and quantization table, we take a similar implementation like [15] to the pre-trained model, giving the loss function as follows:

$$L = \mathcal{F}_{loss}(\boldsymbol{W}) + \frac{\gamma}{2}\|\boldsymbol{W} - f(\boldsymbol{W}) + \boldsymbol{\lambda}\|_2^2 - \frac{\gamma}{2}\|\boldsymbol{\lambda}\|^2 \tag{5}$$

where W denotes the weight in convolution layer, γ represents the penalty parameter and λ is the Lagrangian multiplier, we set $\gamma = 1$ in experiments. The quantization parameters can be solved by minimizing the following:

$$\min_{\alpha, \mathbf{N}} \left\|\boldsymbol{W}^{(k+1)} + \boldsymbol{\lambda}^{(k)} - \alpha \boldsymbol{N}^{(k)}\right\|^2 \tag{6}$$

where N and α can be updated iteratively according to follows:

$$N_{i,\alpha}^{(k+1)} = \arg\min_{N} \left|W_i^{(k+1)} - \alpha N^{(k)}\right| \tag{7}$$

$$\alpha^{(k+1)} = \frac{\left(\boldsymbol{W}_i^{(k+1)} + \boldsymbol{\lambda}^{(k)}\right)^\top \boldsymbol{N}_{i,\alpha}^{(k+1)}}{\boldsymbol{N}_{i,\alpha}^{(k+1)\top} \boldsymbol{N}_{i,\alpha}^{(k+1)}} \tag{8}$$

after N and α converge, λ is calculated as follows:

$$\boldsymbol{\lambda}^{(k+1)} = \boldsymbol{\lambda}^{(k)} + \boldsymbol{W}^{(k+1)} - f\left(\boldsymbol{W}^{(k+1)}\right) \tag{9}$$

The calculation of α and N are realized through multiple rounds of iterations. The quantized weight can be obtained by multiplying α and N obtained in the model. Owing to this multi-iteration nature, we found that the training of ADMM can converge quickly.

3.3. Mixed Precision Quantization

After getting the results of uniform quantization, we seek a more dedicated quantization compression approach. Since each convolution layer in the model is at different depths of the ResNet structure, they have different sensitivity to quantization. A mixed precision quantization approach is proposed by applying different precision quantization to each convolution layer according to their sensitivity. In our work, we evaluate the sensitivity of each convolution layer to quantization compression by computing the trace of Hessian matrix $\boldsymbol{H}$[20], and the total sensitivity of an L-layer ResNet can be calculated as below:

$$\Omega^{\text{Hes}} = \sum_{i=1}^{L} \Omega_i^{\text{Hes}} = \sum_{i=1}^{L} \text{Tr}\left(\boldsymbol{H}_i\right) \cdot \|f\left(\boldsymbol{W}_i\right) - \boldsymbol{W}_i\|_2^2 \tag{10}$$

Since calculating the trace of the Hessian matrix requires a lot of computational overhead, we adopt an approximate method [21] on $\text{Tr}(\boldsymbol{H})$, z_i is a random vector sampled from a Gaussian Distribution $\mathcal{N}(\mathbf{0}, \mathbf{1})$, the trace of Hessian matrix is given as follows:

$$\text{Tr}(\boldsymbol{H}) \approx \frac{1}{m} \sum_{i=1}^{m} \boldsymbol{z}_i^\top \boldsymbol{H} \boldsymbol{z}_i \tag{11}$$

We sort all modules by the trace of their Hessian matrix, the layer with a larger Hessian trace is considered more sensitive to quantization. We ensure that the quantization precision of the model

with high sensitivity is not lower than that of the model with low sensitivity. At the same time, limiting the maximum size of the mixed precision model, a set of mixed precision combinations with the lowest Ω^{Hes} is selected. With the resulting mixed precision quantization configuration, a mixed precision quantization model is generated by initializing the quantization precision of each layer differently. For the convenience of comparison, 8 bit and 6 bit size equivalent mixed precision quantization models are realized in our experiments.

3.4. Batch Normalization Layer Processing in Quantization

In the ResNet structure of speaker verification systems, the convolution layer is always followed by a batch normalization (BN) layer. According to previous studies [6], BN operation can effectively prevent overfitting and speed up training. However, when quantizing the model, the BN layer has become an obstacle that affects the model compression rate. Therefore, we try to merge some of the BN layers in ResNet into the previous convolution layer before the ADMM training in the following way:

$$\alpha = \frac{\gamma}{\sqrt{\sigma^2 + \epsilon}} \tag{12}$$

$$W_{\text{merged}} = W \times \alpha \tag{13}$$

$$B_{\text{merged}} = B \times \alpha + (\beta - \mu \times \alpha) \tag{14}$$

where W_{merged} , B_{merged} are new weight and bias of convolution layer, μ and σ^2 represents the mean value and variance of BN, γ denotes the BN scaling factor, β is the BN offset, ϵ is a very small number. In our experiments, BN layers of different depths are merged independently to evaluate their importance in quantization training.

4. EXPERIMENTAL SETUP

4.1. Datasets

We use the Voxceleb1&2 datasets [22, 23] for our experiments. The training set is the development set of Voxceleb2, and Voxceleb1 is used as testing data. Three official trial lists are used in test sessions to estimate the performance of proposed models. Data augmentation and speed perturbation are applied in the experiments in order to gain model robustness. Noises are added with RIRs [24] and MUSAN [25] to original training utterances. In addition, we adjust speed of the original utterances by a factor of 0.9 and 1.1 to add twice as many speakers to the original dataset.

4.2. Implementation Details

We adopt the state-of-the-art model in speaker verification task: ResNet34 and ResNet152 as the target model of quantization. Different from the previous setting [10], we train model via ADMM for 2 epochs and chunk speech segments of length 200 frames as the training input data of the network. The training loss function is Additive Angular Margin (AAM) loss [26] with an angular margin m of 0.2. The equal error rate (EER) and minimum detection cost function (MinDCF) are referred as performance indicators with the settings of $P_{target} = 0.01$ and $C_{FA} = C_{Miss} = 1$.

5. RESULTS AND ANALYSIS

5.1. Model Quantization Results

In this section, the experimental results of uniform and mixed precision quantization models are analyzed. After 16 bit and 8 bit uniform quantization of the original ResNet model, the performance of the model is basically not affected. The model performance begins to decrease after 6 bit uniform quantization. And after 4 bit uniform quantization, the model accuracy is seriously damaged. Such results can be found from the gap of parameters. As shown in Table 1, when the model is quantized with a higher quantization precision, such as 16 bit and 8 bit, the quantized parameters can basically keep the same as the full-precision parameters($\pm$5.95%). As the compression ratio continues to increase, the gap between the quantized and original parameters becomes larger($\pm$24.11%, $\pm$87.68%). When the input data passes through such layers with significant parameter errors, serious calculation errors will occur, and such errors will gradually accumulate as the number of layers increases. Our experimental results support this interpretation: the 6 bit uniform compression result of ResNet152 is even worse than ResNet34 at the same quantized precision.

Table 1. Partial weight values of the 28-th layer of ResNet34 quantization model and average parameter deviation of the same layer at different quantization precision.

Quantization Precision	Partial Quantized Parameter	Average Deviation
Full-precision (32 bit)	$\begin{bmatrix} 0.0283 & 0.0249 & 0.0180 \\ -0.0222 & -0.0057 & -0.0235 \\ -0.0373 & 0.0006 & 0.0092 \end{bmatrix}$	-
16 bit quantization	$\begin{bmatrix} 0.0283 & 0.0248 & 0.0179 \\ -0.0220 & -0.0055 & -0.0233 \\ -0.0374 & 0.0006 & 0.0091 \end{bmatrix}$	$\pm$0.90%
8 bit quantization	$\begin{bmatrix} 0.0264 & 0.0264 & 0.0198 \\ -0.0231 & -0.0033 & -0.0231 \\ -0.0363 & 0.0000 & 0.0099 \end{bmatrix}$	$\pm$5.95%
6 bit quantization	$\begin{bmatrix} 0.0270 & 0.0270 & 0.0135 \\ -0.0270 & 0.0000 & -0.0135 \\ -0.0405 & -0.0000 & 0.0135 \end{bmatrix}$	$\pm$24.11%
4 bit quantization	$\begin{bmatrix} 0.0512 & 0.0000 & 0.0512 \\ -0.0000 & -0.0000 & -0.0000 \\ -0.0512 & 0.0000 & 0.0000 \end{bmatrix}$	$\pm$87.68%

We found that 8 bit is the maximum compression level without sacrificing the performance of the model. Table 2 shows the results of our quantized model tested on Voxceleb 1. According to the compression results of ResNet34, the EER (%) of the compressed model has a relative increase against to original model of 5.61%, 2.97%, and 3.24%; minDCF is relative increased by 13.6%, 3.1% and 4.6% on Vox1-O, Vox1-E, and Vox1-H, respectively. On ResNet152, 8 bit quantization can save 55% model size, which makes the size of ResNet152 35% larger than that of the original full-precision ResNet34, but obtains a relative 35% improvement in model performance.

Similar to [13], in our experiments, the performance of mixed precision quantization does not outperform uniform quantization models with corresponding sizes. There are two main reasons: firstly, the performance of 8 bit uniform quantization is quite close to the full-precision model, and quantization compression cannot improve the performance of the original model, so there is little room for improvement. The poor performance of the 6 bit mixed precision quantization model can be explained by the poor performance of 4 bit quantization component, who makes it impossible to effectively transmit high-quality data streams throughout the model.

5.2. Analysis of BN Layer in Quantization Training

We focus on the function of batch normalization layers in the quantization training process in this section. In our experiments, any

Table 2. Performance comparison of the full-precision baselines and the proposed quantization compression systems on the Voxceleb1 dataset. "mixed precision 8 bit" represents a mixed precision quantization model with a size comparable to 8 bit uniform quantization. The corresponding quantization precision of each convolution layer of ResNet34 can be found in Figure 1.

Architecture	Quantization precision	Model size	Voxceleb-O		Voxceleb-E		Voxceleb-H	
			EER(%)	MinDCF	EER(%)	MinDCF	EER(%)	MinDCF
ResNet34	32bit(full-precision)	26.7MB	0.89	0.0980	1.01	0.1206	1.85	0.1837
	16 bit	16.0MB	0.90	0.1002	1.03	0.1215	1.88	0.1859
	8 bit	10.7MB	0.94	0.1113	1.04	0.1243	1.91	0.1922
	6 bit	9.4MB	1.73	0.1904	1.83	0.1996	3.32	0.2881
	4 bit	8.1MB	26.82	0.9970	26.5	0.9995	34.72	0.9993
	mixed precision 8 bit	10.7MB	0.93	0.0979	1.08	0.1236	1.96	0.1946
	mixed precision 6 bit	9.4MB	1.82	0.1974	1.95	0.2083	3.47	0.2935
ResNet152	32bit(full-precision)	79.7MB	0.54	0.0496	0.72	0.0787	1.35	0.1263
	16 bit	50.7MB	0.55	0.0493	0.74	0.0812	1.38	0.1295
	8 bit	36.1MB	0.58	0.0546	0.76	0.0847	1.44	0.1361
	6 bit	32.5MB	2.44	0.2643	2.42	0.2356	4.13	0.3281
	4 bit	28.9MB	38.04	0.9981	37.28	0.9990	41.27	0.9991
	mixed precision 8 bit	36.1MB	0.59	0.0979	0.77	0.0847	1.42	0.1328
	mixed precision 6 bit	32.5MB	6.12	0.5042	6.11	0.4712	9.63	0.5809

Table 3. The experiment results of 8 bit uniform quantized ResNet34. BN layers are merged at the beginning of train session. ✓means the layers of ResNet where BN parameters are merged, × means BN layers work independently in corresponding layers.

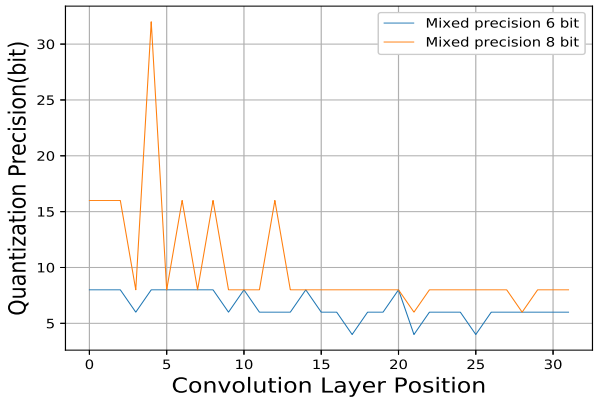
Layer1	Layer2	Layer3	Layer4	Vox1-O EER(%)
×	×	×	×	0.94
×	×	×	✓	5.89
×	×	✓	✓	30.63
✓	✓	×	×	34.12
✓	×	×	×	35.86

transformation of the BN layer before training prevents the whole model from maintaining its performance. Plus, shallower BN layers are more critical to the model training. The results of the BN-merged model are shown in Table 3. Even if we only merge the first few BN layers, this operation has a catastrophic impact on the performance of the entire quantized model, EER of the model with layer1 merged is 35%. In our experiments, the transformation of batch normalization parameters in layer4 causes the least negative impact on performance. Despite this, the corresponding EER still drops to 5.89%. The performance drop reflects the vital role of the BN layer in speaker verification system. The specialized quantization method of normalization layer is necessary to further improve the overall quantization compression rate.

5.3. Quantization Sensitivity Analysis of Convolution Layers

We conduct a discussion of layer sensitivity differences in this section. For the ResNet architecture, the first layers of the network require the highest precision of parameters. As the depth of the network continues to increase, the quantization sensitivity of convolution layers gradually decreases. Our experiment results are shown in Figure 1. Shallower layers are the initial computational units to process the raw data. If the precision of these layers is inadequate, the model performance can hardly be compensated by deeper

Fig. 1. The quantization precision number of each layer of ResNet34 in mixed precision quantization.



convolution layers. Our experimental results also confirm this argument. Therefore, in the case of limited model size after compression, deeper convolution layers should be quantized in the first place.

6. CONCLUSION

In this paper, we apply the alternate direction methods of multipliers (ADMM) to model quantization of speaker verification system. Our results prove that 8 bit is the minimum quantization bit number that ResNet architecture can achieve so far without affecting the performance. In addition, our study verifies the importance of batch normalization layers in quantization training through related ablation studies. Compared with the original model, the quantized model using ADMM can keep the performance drop by less than 6% under the condition of about 55% compression in experiments on Voxceleb. We also find the shallower convolution layers are more sensitive to quantization and the batch normalization is indispensable to quantization training. In future work, we will aim to maintain the performance of quantization models at high compression ratios.

7. REFERENCES

[1] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2017, pp. 999–1003.

[2] Y. Liu, Y. Qian, N. Chen, T. Fu, Y. Zhang, and K. Yu, "Deep feature for text-dependent speaker verification," *Speech Communication*, vol. 73, pp. 1–13, 2015.

[3] B. Liu, Z. Chen, and Y. Qian, "Dual Path Embedding Learning for Speaker Verification with Triplet Attention," in *Proc. Interspeech 2022*, 2022, pp. 291–295.

[4] B. Desplanques, J. Thienpondt, and K. Demuynck, "Ecapa-tdnn: emphasized channel attention, propagation and aggregation in tdnn based speaker verification," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2020, pp. 3830–3834.

[5] B. Liu, Z. Chen, S. Wang, H. Wang, B. Han, and Y. Qian, "DF-ResNet: Boosting Speaker Verification Performance with Depth-First Design," in *Proc. Interspeech 2022*, 2022, pp. 296–300.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[7] Y. Izui and A. Pentland, "Analysis of neural networks with redundancy," *Neural Computation*, vol. 2, no. 2, pp. 226–238, 1990.

[8] Y. Cheng, F. Yu, R. Feris, S. Kumar, A. Choudhary, and S. Chang, "An exploration of parameter redundancy in deep networks with circulant projections," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2857–2865.

[9] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[10] B. Liu, H. Wang, Z. Chen, S. Wang, and Y. Qian, "Self-knowledge distillation via feature enhancement for speaker verification," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 7542–7546.

[11] S. Wang, Y. Yang, T. Wang, Y. Qian, and K. Yu, "Knowledge distillation for small foot-print deep speaker embedding," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6021–6025.

[12] J. Luo, J. Wu, and W. Lin, "Thinet: A filter level pruning method for deep neural network compression," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5058–5066.

[13] J. Xu, S. Hu, X. Liu, and H. Meng, "Towards green asr: Lossless 4-bit quantization of a hybrid tdnn system on the 300-hr switchboard corpus," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2022.

[14] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "Xnor-net: Imagenet classification using binary convolutional neural networks," in *European conference on computer vision*. Springer, 2016, pp. 525–542.

[15] J. Xu, X. Chen, S. Hu, J. Yu, X. Liu, and H. Meng, "Low-bit quantization of recurrent neural network language models using alternating direction methods of multipliers," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7939–7943.

[16] C. Leng, Z. Dou, H. Li, S. Zhu, and R. Jin, "Extremely low bit neural network: Squeeze the last bit out with admm," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[17] T. Zhu, X. Qin, and M. Li, "Binary neural network for speaker verification," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2021, pp. 86–90.

[18] J. Xu, J. Yu, X. Liu, and H. Meng, "Mixed precision dnn quantization for overlapped speech separation and recognition," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7297–7301.

[19] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[20] Z. Dong, Z. Yao, D. Arfeen, A. Gholami, M. Mahoney, and K. Keutzer, "Hawq-v2: Hessian aware trace-weighted quantization of neural networks," *Advances in neural information processing systems*, vol. 33, pp. 18518–18529, 2020.

[21] J. Xu, J. Yu, S. Hu, X. Liu, and H. Meng, "Mixed precision low-bit quantization of neural network language models for speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3679–3693, 2021.

[22] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2017, pp. 2616–2620.

[23] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: deep speaker recognition," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2018, pp. 1086–1090.

[24] T. Ko, V. Peddinti, D. Povey, M. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5220–5224.

[25] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[26] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4690–4699.