INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Re-weighted Interval Loss for Handling Data Imbalance Problem of End-to-End Keyword Spotting

Kun Zhang[1], *Zhiyong Wu*[1,2,3,*], *Daode Yuan*[4], *Jian Luan*[4], *Jia Jia*[1,2], *Helen Meng*[1,3], *Binheng Song*[1]

[1]Tsinghua-CUHK Joint Research Center for Media Sciences, Technologies and Systems, Graduate School at Shenzhen, Tsinghua University, Shenzhen, China
[2]Beijing National Research Centre for Information Science and Technology (BNRist), Department of Computer Science and Technology, Tsinghua University, Beijing, China
[3]Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong SAR, China
[4]Xiaoice, Microsoft, Beijing, China

zk17@mails.tsinghua.edu.cn, {zywu, songbinheng}@sz.tsinghua.edu.cn
{Daode.Yuan, jianluan}@microsoft.com, jjia@tsinghua.edu.cn, hmmeng@se.cuhk.edu.hk

Abstract

The training process of end-to-end keyword spotting (KWS) suffers from critical data imbalance problem that positive samples are far less than negative samples where different negative samples are not of equal importances. During decoding, false alarms are mainly caused by a small number of *important negative samples* having pronunciation similar to the keyword; however, the training loss is dominated by the majority of negative samples whose pronunciation is not related to the keyword, called *unimportant negative samples*. This inconsistency greatly degrades the performance of KWS and existing methods like focal loss don't discriminate between the two kinds of negative samples. To deal with the problem, we propose a novel re-weighted interval loss to re-weight sample loss considering the performance of the classifier over local interval of negative utterance, which automatically down-weights the losses of unimportant negative samples and focuses training on important negative samples that are prone to produce false alarms during decoding. Evaluations on Hey Snips dataset demonstrate that our approach has yielded a superior performance over focal loss baseline with 34% (@0.5 false alarm per hour) relative reduction of false reject rate.

Index Terms: keyword spotting, end-to-end, data imbalance, re-weighting, speech recognition

1. Introduction

Keyword spotting (KWS) aims at detecting a pre-defined keyword from a stream of audio. Most voice interface-based smart devices rely on KWS techniques to start human-machine interactions (e.g. "Okay Google" for Android, "Hi Siri" for iPhone). Early researches exploit speech recognition techniques such as large vocabulary continuous speech recognition (LVCSR) [1, 2], keyword/filler hidden Markov models (HMMs) [3, 4] for KWS. With the great breakthroughs of deep learning, KWS models based on deep neural networks (DNNs) have been proposed, including DNNs [5, 6, 7, 8], convolutional neural networks (CNNs) [9], time delay neural networks (TDNNs) [10, 11]. The neural networks read a narrow input window and predict posteriors of sub-keyword (syllable, word, etc.) and filler (non-keyword speech) targets, called **multi-class** models. The posterior handling is further used to produce a confidence score and the system triggers if the confidence score exceeds a preset threshold. The Deep KWS above has significant outperformance over LVCSR and HMM-based methods. However, using sub-keyword targets requires a well-trained acoustic model to obtain frame-level alignments for labeling and complicated posterior handling for decoding. Recently several **end-to-end** KWS models [12, 13] have been proposed, which directly predict the posteriors of binary targets (complete keyword or filler) in an end-to-end manner without necessity for forced-alignment labeling and complicated posterior handling.

However, the training of end-to-end keyword spotting (KWS) suffers from critical data imbalance problem that positive samples are far less than negative samples [14]. Models trained on this data distribution perform poorly on the class with fewer samples. The most effective solution is re-weighing the training loss, also called cost-sensitive learning [15], which can be divided into two regimes: **class balanced re-weighting** and **sample importance re-weighting**. Formally, the loss function of a sample x with label y can be re-weighted by class balanced weight W_c and sample importance weight W_s simultaneously:

$$L_{RE} = W_c \cdot W_s \cdot L(\hat{y}, y) \tag{1}$$

where $\hat{y}$ is the prediction of the model and $L(\hat{y}, y)$ is the raw classification loss (e.g. cross-entropy). For class balanced re-weighting, the weight value W_c is calculated according to the label of samples. The most common practice is setting W_c to inverse class frequency or inverse square root of class frequency. A more advanced approach is proposed in [16] to quantize the effective number of samples in certain class for class balanced re-weighting.

In this work, we focus on sample importance re-weighting to handle the data imbalance problem of end-to-end KWS. While class balanced re-weighting balances the importance of positive/negative samples according to the number of samples of each class, sample importance re-weighting discriminates between samples of different importance considering the sample difficulty (the performance of the classifier on the sample). A typical instance of sample importance re-weighting is focal loss, which is first proposed to address the data imbalance problem of dense object detection [17] and later applied to multi-class KWS model [18]. By adding a dynamically scaled factor to the cross-entropy loss, focal loss can automatically

* Corresponding author

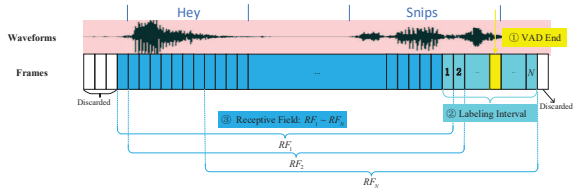
http://dx.doi.org/10.21437/Interspeech.2020-1644

down-weight gradient backpropagation of well-classified samples during training and focus the model on hard samples. As for end-to-end KWS, during decoding, false alarms are mainly caused by a small number of negative samples having pronunciation similar to the keyword (**important negative samples**), which cause continuous false positive predictions; however, the training loss of end-to-end KWS is dominated by the majority of negative samples producing discrete false positive predictions, of which the pronunciation is not related to the keyword (**unimportant negative samples**). This inconsistency greatly degrades the performance of KWS. Focal loss doesn't discriminate between them because it uses frame loss as re-weighting unit to calculates sample importance weight according to *frame sample difficulty*.

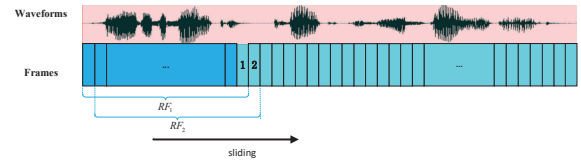
To deal with the problem, we propose a novel re-weighted interval loss to re-weight sample loss considering *local sample difficulty* (the performance of the classifier over samples within a local interval of negative utterance), which automatically down-weights the losses of unimportant negative samples and focus training on important negative samples which are prone to produce false alarms during decoding.

2. Related Work and Problem Analysis

2.1. Interval labeling mechanism



(a) Interval Labeling for Positive Utterance



(b) Sliding Labeling for Negative Utterance

Figure 1: *Labeling mechanism for positive utterance and negative utterance.*

The most important feature of end-to-end KWS is the use of Voice Activity Detection (VAD) technique to decide the boundary of positive sample rather than costly forced-alignment with well-trained acoustic model. In [12], a labeling interval around VAD end of the keyword is adopted to deal with the VAD detection error. Specifically, take keyword "Hey Snips" as an example, the labeling of positive utterance consists of three steps as shown in Figure 1(a). First, VAD is used to decide the end of keyword (**VAD end**). Second, positive label is assigned to frames within a **labeling interval** of length N around the VAD end to make sure the **receptive field** or input window of the KWS model covers the complete keyword speech. Frames outside of the labeling interval are not considered. Finally, N Receptive Field (from RF_1 sliding to RF_N) ending with the labeled frames within the labeling interval constitute the N positive samples of the positive utterance. Due to the interval labeling above, the model learns a pattern of keyword speech with time shift invariance within labeling interval, meaning the model tends to make continuous true positive predictions around the end of well-classified keyword speech.

As for the labeling of negative utterance, receptive field can be slided along the timeline, as shown in Figure 1(b). Same as positive samples, the frame at the end of the receptive field are labeled as negative. For the convenience of description, we refer to the classification loss of the receptive field as **frame loss**.

2.2. Problem Analysis

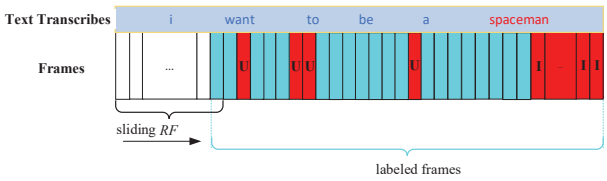


Figure 2: *Important ("I") and unimportant ("U") negative samples of end-to-end KWS*

During decoding of KWS, the false alarms are mainly caused by a small number of important negative samples having pronunciation similar to the keyword. When important negative samples are mistaken for keyword by the model, continuous false positive predictions are produced around the end of the speech like the true positive case. Take the negative utterance ("i want to be a spaceman") as an example as shown in Figure 2. The pronunciation of "**s**pa**ce**man" is similar to that of the keyword "hey **s**nip**s**", which makes it an important negative sample causing continuous false positive predictions around its end (red frames labeled with "I" in Figure 2). The pronunciations of unimportant negative samples like "i want to be a" are not related to that of keyword, and may produce occasional discrete false positive predictions (red frames labeled with "U" in Figure 2) due to the reason that outputs of neural network model are not smooth with respect to its inputs [19, 20]. During training, unimportant negative samples constitute the majority of the loss and dominate the gradient.

The "I" and "U" false positive predictions caused by important and unimportant negative samples respectively in Figure 2 are not differentiated by the focal loss[17, 18], since it treats all such false positive frames equally and uses single frame loss as re-weighting unit to calculate W_s considering the frame sample difficulty:

$$FL(p_t) = (1-p_t)^{\gamma} \cdot -\log p_t \tag{2}$$

in which p_t is the posterior corresponding to the groundtruth output by the model, $-\log p_t$ is the standard cross-entropy loss and $(1-p_t)^{\gamma}$ is the modulating factor, i.e. sample importance weight W_s. When the sample is well-classified (blue frames in Figure 2), p_t approaches to 1, hence the sample importance weight $(1-p_t)^{\gamma}$ is close to 0 and the loss is down-weighted. For misclassified frames (the red frames including both "I" and "U" in Figure 2), p_t approaches to 0, hence the sample importance weight $(1-p_t)^{\gamma}$ goes to 1 and the loss is less affected.

3. Methodology

In this section, we introduce a novel re-weighted interval loss to automatically down-weight the losses of unimportant negative samples and focus training on important negative samples.

Rather than using frame loss as re-weighting unit like focal loss, re-weighted interval loss merges frame losses within the labeling interval into single interval loss and re-weights the interval loss considering the local sample difficulty, i.e. the proportion of false positive predictions within the labeling interval.

3.1. Interval loss

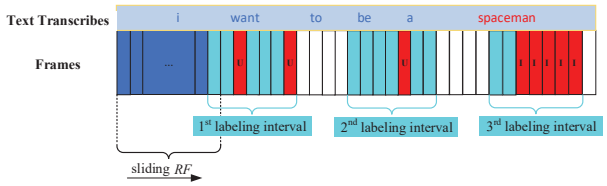


Figure 3: *Sample importance re-weighting base on local sample difficulty*

We label the negative utterance in a way similar to the positive utterance as shown in Figure 3. The receptive field slides over frames within the labeling interval of length N, and the last frame of the receptive field is labeled as negative. The spacing between adjacent labeling intervals helps to down-sample the negative samples. There may be multiple labeling intervals for single negative utterance because it's usually much longer than the keyword. For each labeling interval of the negative utterance, we merge N frame losses, i.e. the classification losses of N receptive fields, into single **interval loss** L_I by average pooling (ave-pooling):

$$L_I = \frac{1}{N}\sum_{i=1}^{N} L(\hat{y}_i, y_i) \tag{3}$$

where $L(\hat{y}_i, y_i)$ is the cross-entropy loss of i-th receptive field.

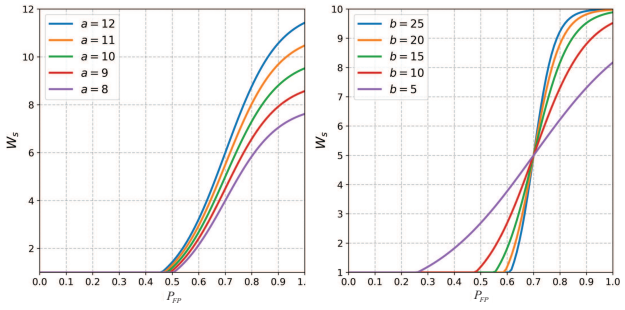
3.2. Sample importance re-weighting based on local sample difficulty

We use interval loss above as re-weighting unit instead of single frame loss in focal loss. The sample importance weight of the interval loss is calculated considering the local sample difficulty, i.e. the performance of the classifier over local interval of negative utterance, which is formally represented by the proportion of false positive predictions (red frames in Figure 3) in all N frames of the labeling interval. The false positive prediction means that the output positive posterior probability $P\{y = 1|x\}$ of the receptive field exceeds 0.5 while the last frame of the receptive field is labeled as negative. Specifically, the number of false alarm predictions within the labeling interval is N_{FP}, and we have the proportion of false positive predictions $P_{FP} = \frac{N_{FP}}{N}$. The sample importance weight W_s of the interval loss L_I is as follows:

$$W_s = max\{1, \frac{a}{1 + e^{-b(P_{FP} - P_T)}}\} \tag{4}$$

where P_T is the proportion threshold, a is the upper bound of the weight and b decides the function gradient around proportion threshold P_T. The visualization of sample importance weight function W_s ($P_T = 0.7$) is shown in Figure 4, which demonstrates the effects of the hyparameters a and b.

Take the negative utterance ("i want to be a spaceman") as an example as shown in Figure 3. The 1^{st} labeling interval and the 2^{nd} labeling interval covers only unimportant negative samples ("i want to be a"), where the proportion of false positive predictions (red frames labeled with "U") is small and the interval loss is down-weighted. The 3^{rd} labeling interval covers important negative samples ("spaceman"), where the proportion of false positive predictions (red frames labeled with "I") is large and the interval loss is less affected. In this way, our approach can discriminate the two kinds of negative samples by differencing the importance between them.



(a) (b)

Figure 4: *The visualization of sample importance weight function W_s ($P_T = 0.7$) derived from sigmoid function: (a) a is the upper bound of the weight; (b) b decides the function gradient around proportion threshold P_T.*

3.3. Re-weighted interval loss

The final re-weighted interval loss L_{RE} of negative samples is as follows:

$$L_{RE} = W_c \cdot max\{1, \frac{a}{1 + e^{-b(P_{FP} - P_T)}}\} \cdot L_I \tag{5}$$

where W_c is the class balanced weight. As for positive samples, the sample importance weight W_s is set to 1, and corresponding re-weighted interval loss is $L_{RE} = W_c \cdot L_I$.

4. Experiment

4.1. Data

We evaluate our re-weighted interval loss on an open dataset [12] whose keyword is "Hey Snips". The dataset consists of about 11K keyword utterances and 86.5K (96 hours) negative non-keyword utterances. Negative utterances have been collected in the same conditions (speaker, hardware, environment, etc.) with keyword utterances. The acoustic features are 20-dimensional log-Mel filterbank energies (LFBEs), which is extracted from the input audio every 10ms over a window of 25ms.

4.2. Experimental setup

We choose two end-to-end KWS models based on CNNs to evaluate our proposed method: *dilated CNNs* [12] and *trad fpool3 CNNs* [9]. Our experiments are conducted based on TensorFlow and ADAM optimizer [21] with a learning rate of 10^{-3} and a batch size of 256. We impose a class balanced re-weighting with the same weight ratio ($Positive : Negative = 10 : 1$) on all the compared methods. All the hyperparameters are tuned on the dev set ($a = 10$, $b = 10$, $P_T = 0.7$, $N = 31$). The compared methods are described as follows:

1) **Cross-Entropy Loss (CEL)**: cross-entropy loss with no sample importance re-weighting;

2) **Focal Loss (FL)**: focal loss in [18] ($\alpha = 0.5, \gamma = 1$);

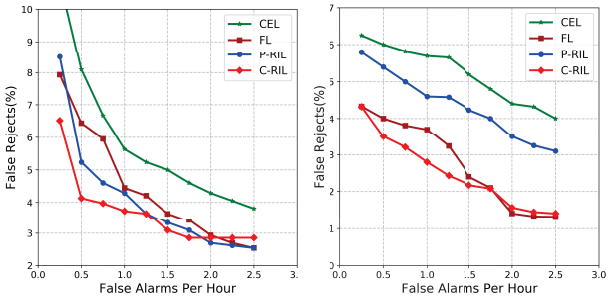
3) **Continuous Re-weighted Interval Loss (C-RIL)**: re-weighed interval loss with continuous weight function in Equation (4);

4) **Piecewise Re-weighted Interval Loss (P-RIL)**: re-weighted interval loss with piecewise weight function in Equation (6), by approximating $b \to \infty$:

$$W_s = \begin{cases} W_1, & P_{FP} \geq P_T, \\ W_2, & P_{FP} < P_T \end{cases} \tag{6}$$

where W_1 and W_2 are upper and lower bounds respectively which are tuned on dev set ($W_1 = 10$ and $W_2 = 1$).

4.3. Experimental results



(a) dilated CNNs (b) trad fpool3 CNNs

Figure 5: *DET curves for different model architectures*

Figure 5 provides the Detection Error Tradeoff (DET) curves of *dilated CNNs* and *trad fpool3 CNNs* for all the compared loss functions. We can see from the curves that C-RIL outperforms other baselines at almost all the values of the threshold, which proves that re-weighted interval loss helps model focus on important negative samples and improves significantly the performance of end-to-end KWS under the data imbalance condition. Additionally, it can be seen that FL has yielded a superior performance over CEL but is still weaker than C-RIL because using frame as re-weighting unit and it doesn't discriminate between important and unimportant negative samples during training.

We evaluate False Reject Rate (FRR) by tuning threshold to fix the False Alarms at 0.5 and 1.0 per hour, as shown in Table 1. The proposed C-RIL performs the best and yields a lower FRR than FL baseline with a 34% (@0.5 False Alarm per hour) and 16% relative reduction on *dilated CNNs*. Another interesting observation is that continuous weight function (C-RIL) is better than the piecewise one (P-RIL), we guess the reason is continuous weight function makes sample importance weight W_s more differentiated with respect to the local sample difficulty P_{FP} The result also demonstrates that C-RIL on *trad fpool3 CNNs* performs better than C-RIL on *dilated CNNs*, we think the reason is the former has used more parameters and calculations.

4.4. Methods of merging losses: ave-pooling vs. max-pooling

In our method, we have proposed to use ave-pooling to calculate the interval loss as in Equation (3). Another way to merge frame losses into interval loss is max-pooling inspired by [22]. Formally, we can get the merged interval loss L_I by max-pooling:

$$L_I = \max_{1 \leq i \leq N} L(\hat{y_i}, y_i) \tag{7}$$

where $L(\hat{y_i}, y_i)$ is the cross-entropy loss of i-th receptive field. The DET curves of re-weighted interval loss (both C-RIL and P-RIL) using max-pooling and ave-pooling on *dilated CNNs* is shown in Figure 6. It's clear that ave-pooling significantly outperforms max-pooling in both C-RIL and P-RIL. The possible reason is merging frame losses by max-pooling discards all other frame losses within the labeling interval except the maximum, which makes model under-fitting. We also find that max-pooling method greatly slows the convergence of training process since there is only one frame loss left for the back propagation of gradients.

Table 1: *False Reject Rate (FRR) (%) calculated at 0.5/1.0 false alarm per hour*

0.5/1.0 false alarm	**CEL**	**FL**	**P-RIL**	**C-RIL**
dilated CNNs	8.09/5.63	6.42/4.44	5.07/4.20	**4.21/3.73**
trad fpool3 CNNs	6.07/5.71	4.05/3.73	5.48/4.62	**3.53/2.82**

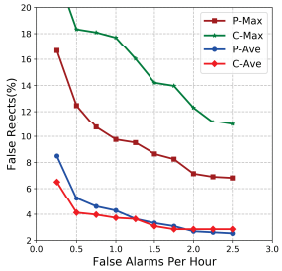


Figure 6: *P-Max (P-RIL with Max-pooling), C-Max (C-RIL with Max-pooling), P-Ave (P-RIL with Ave-pooling) and C-Ave (C-RIL with Ave-pooling).*

5. Conclusion

In this paper, we explore the re-weighted interval loss for handling the data imbalance problem of end-to-end KWS. The proposed re-weighted loss is intended to use the interval loss as re-weighting unit and calculates sample importance weight considering the local sample difficulty. Evaluations on Hey Snips dataset demonstrate that our approach has yielded a lower FRR than focal loss baseline.

6. Acknowledgements

This work was conducted when the first author was an intern at Microsoft, and is supported by joint research fund of National Natural Science Foundation of China - Research Grant Council of Hong Kong (NSFC-RGC) (61531166002, N_CUHK404/15), National Natural Science Foundation of China (61521002, 61433018, 61375027). We would also like to thank Tencent AI Lab Rhino-Bird Focused Research Program (No. JR202041, JR201942) and Tsinghua University - Tencent Joint Laboratory for the support.

7. References

[1] D. R. Miller, M. Kleber, C.-L. Kao, O. Kimball, T. Colthurst, S. A. Lowe, R. M. Schwartz, and H. Gish, "Rapid and accurate spoken term detection," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2007, pp. 314–317.

[2] J. Mamou, B. Ramabhadran, and O. Siohan, "Vocabulary independent spoken term detection," in *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2007, pp. 615–622.

[3] J. R. Rohlicek, W. Russell, S. Roukos, and H. Gish, "Continuous hidden markov modeling for speaker-independent word spotting," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1989, pp. 627–630.

[4] R. C. Rose and D. B. Paul, "A hidden markov model based keyword recognition system," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1990, pp. 129–132.

[5] G. Chen, C. Parada, and G. Heigold, "Small-footprint keyword spotting using deep neural networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 4087–4091.

[6] S. Panchapagesan, M. Sun, A. Khare, S. Matsoukas, A. Mandal, B. Hoffmeister, and S. Vitaladevuni, "Multi-task learning and weighted cross-entropy for dnn-based keyword spotting," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2016, pp. 760–764.

[7] Y. Yuan, Z. Lv, S. Huang, and L. Xie, "Verifying deep keyword spotting detection with acoustic word embeddings," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 613–620.

[8] J. Hou, Y. Shi, M. Ostendorf, M.-Y. Hwang, and L. Xie, "Region proposal network based small-footprint keyword spotting," vol. 26, no. 10, 2019, pp. 1471–1475.

[9] T. N. Sainath and C. Parada, "Convolutional neural networks for small-footprint keyword spotting," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2015, pp. 1478–1482.

[10] M. Sun, D. Snyder, Y. Gao, V. K. Nagaraja, M. Rodehorst, S. Panchapagesan, N. Strom, S. Matsoukas, and S. Vitaladevuni, "Compressed time delay neural network for small-footprint keyword spotting," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2017, pp. 3607–3611.

[11] S. Myer and V. S. Tomar, "Efficient keyword spotting using time delay neural networks," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2018, pp. 1264–1268.

[12] A. Coucke, M. Chlieh, T. Gisselbrecht, D. Leroy, M. Poumeyrol, and T. Lavril, "Efficient keyword spotting using dilated convolutions and gating," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6351–6355.

[13] C. Shan, J. Zhang, Y. Wang, and L. Xie, "Attention-based end-to-end models for small-footprint keyword spotting," in *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2018, pp. 2037–2041.

[14] J. Hou, Y. Shi, M. Ostendorf, M.-Y. Hwang, and L. Xie, "Mining effective negative training samples for keyword spotting," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7444–7448.

[15] C. Elkan, "The foundations of cost-sensitive learning," in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2001, pp. 973–978.

[16] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 9268–9277.

[17] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988.

[18] B. Liu, S. Nie, Y. Zhang, S. Liang, Z. Yang, and W. Liu, "Focal loss and double-edge-triggered detector for robust small-footprint keyword spotting," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6361–6365.

[19] X. Wang, S. Sun, C. Shan, J. Hou, L. Xie, S. Li, and X. Lei, "Adversarial examples for improving end-to-end attention-based small-footprint keyword spotting," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6366–6370.

[20] X. Wang, S. Sun, and L. Xie, "Virtual adversarial training for ds-cnn based small-footprint keyword spotting," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 607–612.

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *arXiv preprint arXiv:1412.6980*, 2014.

[22] M. Sun, A. Raju, G. Tucker, S. Panchapagesan, G. Fu, A. Mandal, S. Matsoukas, N. Strom, and S. Vitaladevuni, "Max-pooling loss training of long short-term memory networks for small-footprint keyword spotting," in *IEEE Spoken Language Technology Workshop (SLT)*, 2016, pp. 474–480.