한국음향학회지 제39권 제5호 pp. 454~460 (2020)
The Journal of the Acoustical Society of Korea Vol.39, No.5 (2020)
https://doi.org/10.7776/ASK.2020.39.5.454

pISSN : 1225-4428
eISSN : 2287-3775

다양한 합성곱 신경망 방식을 이용한 모바일 기기를 위한 시작 단어 검출의 성능 비교

Performance comparison of wake-up-word detection on mobile devices using various convolutional neural networks

김상홍,[1] 이보원[1†]

(Sanghong Kim[1] and Bowon Lee[1†])

[1]인하대학교 전자공학과

(Received May 21, 2020; accepted July 7, 2020)

초 록: 음성인식 기능을 제공하는 인공지능 비서들은 정확도가 뛰어난 클라우드 기반의 음성인식을 통해 동작한다. 클라우드 기반의 음성인식에서 시작 단어 인식은 대기 중인 기기를 활성화하는 데 중요한 역할을 한다. 본 논문에서는 공개 데이터셋인 구글의 Speech Commands 데이터셋을 사용하여 스펙트로그램 및 멜-주파수 캡스트럼 계수 특징을 입력으로 하여 모바일 기기에 대응한 저 연산 시작 단어 검출을 위한 합성곱 신경망의 성능을 비교한다. 본 논문에서 사용한 합성곱 신경망은 다층 퍼셉트론, 일반적인 합성곱 신경망, VGG16, VGG19, ResNet50, ResNet101, ResNet152, MobileNet이며, MobileNet의 성능을 유지하면서 모델 크기를 1/25로 줄인 네트워크도 제안한다.
핵심용어: 성능 비교, 시작 단어 검출, 합성곱 신경망, 인공지능 비서

ABSTRACT: Artificial intelligence assistants that provide speech recognition operate through cloud-based voice recognition with high accuracy. In cloud-based speech recognition, Wake-Up-Word (WUW) detection plays an important role in activating devices on standby. In this paper, we compare the performance of Convolutional Neural Network (CNN)-based WUW detection models for mobile devices by using Google's speech commands dataset, using the spectrogram and mel-frequency cepstral coefficient features as inputs. The CNN models used in this paper are multi-layer perceptron, general convolutional neural network, VGG16, VGG19, ResNet50, ResNet101, ResNet152, MobileNet. We also propose network that reduces the model size to 1/25 while maintaining the performance of MobileNet is also proposed.
Keywords: Performance comparison, Wake-up-word detection, Convolutional neural network, Artificial Intelligence (AI) assistant

PACS numbers: 43.71.Ft, 43.72.Bs

I. 서 론

최근 인공지능 기술의 비약적인 발전과 더불어 음성 인터페이스를 기반으로 하는 음성인식 및 음성합성 분야가 대두되고 있으며 사용자와의 실시간 대화가 가능한 인공지능 비서가 적용된 기기들이 대중적으로 보급되고 있다.

인공지능 비서가 적용된 기기들은 상대적으로 저성능 처리 장치를 사용하기 때문에 음성인식 기술을 자체적으로 동작시키기에는 처리 속도에 문제가 있다. 또한, 인공지능 비서의 경우 즉각적인 반응을 위해서 항상 저전력으로 음성인식기를 작동시켜야 한다. 이러한 문제를 해결하기 위해서 인공지능 비서가 적용된 기기들에서 문장에 대한 음성인식을

†**Corresponding author:** Bowon Lee (bowon.lee@dsp.inha.ac.kr)
Department of Electronic Engineering, Inha University, 100 Inha-ro, Michuhol-gu, Incheon 22212, Republic of Korea
(Tel: 82-32-860-7423, Fax: 82-32-868-3654)

진행하는 것이 아닌 “시작”, “안녕”과 같은 작은 단위의 단어를 검출하는 시작 단어 검출(wake-up-word detection)을 통해 기기를 동작시킨 뒤 클라우드 서버를 이용한 음성인식을 진행하여 복잡한 문장의 문장 인식 속도를 향상 시킨다. 음성인식 기능이 있는 인공지능 비서를 저전력으로 구현시키기 위해서는 시작 단어 검출 알고리즘의 복잡도가 낮아야 한다.

기존의 시작 단어 검출을 위한 알고리즘으로 통계학적 모델인 은닉 마르코프 모델(Hidden Markov Model, HMM),[1] 기계학습 알고리즘인 서포트 벡터 머신(Support Vector Machine, SVM)[2]을 이용하였다. 최근에는 다층 퍼셉트론(Multi Layer Perceptron, MLP), 합성곱 신경망(Convolutional Neural Network, CNN),[3] 및 순환 신경망(Recurrent Neural Network, RNN) 등의 딥러닝 네트워크를 이용하여 시작 단어 검출을 진행하는 추세이다. 시작 단어 검출의 성능을 향상시키기 위해서는 더 깊은 층을 가진 딥러닝 네트워크를 사용하는 것이 정확도 향상에는 유리하지만, 네트워크의 층을 더 깊게 쌓을수록 알고리즘의 복잡도, 즉 연상량이 증가하는 단점이 있다. 층이 깊어질수록 다음 합성곱의 입력으로 들어가는 채널의 수가 증가하게 되고 이때 합성곱에 대한 파라미터의 수가 기하급수적으로 증가하기 때문이다.

파라미터의 수가 많아지면 인공지능 비서에서 합성곱 신경망을 이용한 시작 단어 검출이 어려워진다. 이러한 문제를 해결하기 위해 합성곱의 연산량 및 채널의 수, 입력의 크기를 조절하는 방식의 합성곱 신경망에 관한 연구가 활발히 진행되고 있다.[4-6]

본 논문에서는 시작 단어 검출을 위하여 입력에 대한 스펙트로그램 특징과 멜-주파수 캡스트럼 계수(Mel-Frequency Cepstral Coefficient, MFCC)[7] 특징 추출을 진행한다. 추출된 특징을 이용하여 다양한 합성곱 신경망들을 이용한 시작 단어 검출 성능을 비교한다. 또한, 최근에 제안된 MobileNet[5]의 구조를 기반으로 하여 정확도는 유사하면서 모델 크기를 1/25로 줄인 네트워크(reduced MobileNet, RMN)를 제안한다. 제안한 네트워크는 기존 네트워크와 유사한 성능을 보이면서 모델 크기를 획기적으로 줄였으므로 저전력 기기에서 시작 단어 인식을 수행하기에 적합하다.

본 논문의 구성은 다음과 같다. II장에서는 모바일 기기에서 시작 단어 검출을 위한 합성곱 신경망을 설명하고 III장에서는 실험 환경, 시작 단어 검출을 위한 데이터, 합성곱 신경망 성능평가 방법 및 합성곱 신경망을 이용한 시작 단어 검출의 결과를 설명하며 마지막으로 IV장에서는 결론과 향후 연구 방향을 설명한다.

II. 모바일 기기에서 시작 단어 검출을 위한 합성곱 신경망

Fig. 1은 시작 단어 검출을 위한 기본 구조를 보여준다. 본 논문에서는 wav 파일에 대해서 MFCC 또는 스펙트로그램으로 특징을 추출한 96 × 40의 2차원 이미지를 입력으로 하여 심층 신경망 기반의 다양한 분류를 진행하였다. 심층 신경망의 출력은 검출된 시작 단어이다.

본 논문에서는 시작 단어 검출을 위해 MLP, CNN, VGGNet,[8] ResNet,[9] MobileNet-V1(MN1)[5] 및 V2(MN2)[6]를 비교하고, 파라미터 수를 1/25로 줄인 RMN을 제안한다.

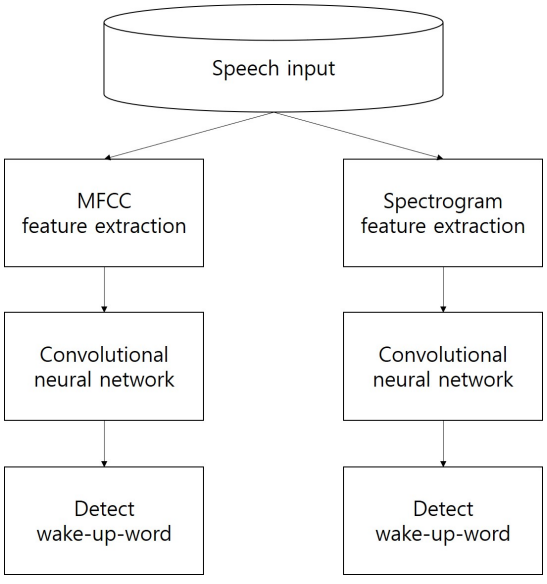

Fig. 1. Architecture of the basic framework for wake-up-word detection.

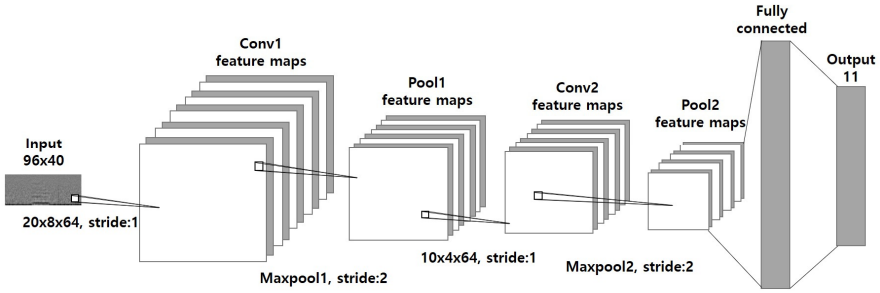

Fig. 2. Architecture of the convolutional neural network.

2.1 MLP

본 논문에서 시작 단어 검출의 성능 비교를 위해 사용한 MLP는 한 개의 입력 층, 총 5개의 은닉층과 한 개의 출력층으로 총 7개의 층으로 구성되어 있다.

본 논문에서 사용한 MLP의 5개의 은닉층은 각각 총 128개의 노드로 구성되어 있으며 학습 과정에서 드롭아웃을 사용하여 정칙화를 하였다. 각 노드에서 사용한 활성 함수는 Rectified Linear Unit(ReLU)이다.

2.2 CNN

본 논문에서 시작 단어 검출의 성능 비교를 위해 사용한 합성곱 신경망은 2개의 합성곱 층과 2개의 풀링 층 및 1개의 완전연결 층을 구성하였다. 본 논문에서 사용한 합성곱 신경망의 구조는 Fig. 2과 같고 각 필터에 대한 설명은 Table 1과 같다.

첫 번째 합성곱 필터는 $20 \times 8 \times 64$를 사용하였으며 두 번째 합성곱 필터는 $10 \times 4 \times 64$를 사용하였고 11개의 범주에 대응하는 완전 연결 층을 이용하였다. 각 합성곱 이후 Max-pooling을 진행하였고 활성 함수는 ReLU를 이용하였다.

2.3 VGGNet

VGGNet[8]은 CNN을 깊게 쌓은 구조로 2014년 ImageNet Challenge에서 준우승을 차지한 모델이다. Table 1은 VGG16과 VGG19의 네트워크 구조를 보여준다.

2.4 ResNet

ResNet은 일반적인 합성곱의 가중치 정보만을 이용하는 것이 아니라 이전 입력의 정보를 연결 생략 개념을 도입하여 합성곱 결과와 합성곱을 진행하기 이전의 결과를 합하는 것을 이용한 네트워크로 이 과정에서 처음에 들어오는 입력은 채널이 많은 형태이고 1×1 합성곱을 이용하여 채널을 줄여 다음 층에서 병목 구조를 만드는 과정을 거친다.

ResNet의 기본 구조는 Table 2와 같다. 여기서 Conv4_x 층의 N에 해당하는 숫자가 6이면 ResNet50, 23이면 ResNet101, 36이면 ResNet152이다.

Table 1. The structure of VGG Networks.

Layer	VGG16	VGG19
Input	224x224x3	224x224x3
Conv1_x	$\begin{bmatrix}3\times3, 64\\3\times3, 64\end{bmatrix}$	$\begin{bmatrix}3\times3, 64\\3\times3, 64\end{bmatrix}$
Pool1	maxpool	maxpool
Conv2_x	$\begin{bmatrix}3\times3, 128\\3\times3, 128\end{bmatrix}$	$\begin{bmatrix}3\times3, 128\\3\times3, 128\end{bmatrix}$
Pool2	maxpool	maxpool
Conv3_x	$\begin{bmatrix}3\times3, 256\\3\times3, 256\\3\times3, 256\end{bmatrix}$	$\begin{bmatrix}3\times3, 256\\3\times3, 256\\3\times3, 256\\3\times3, 256\end{bmatrix}$
Pool3	maxpool	maxpool
Conv4_x	$\begin{bmatrix}3\times3, 512\\3\times3, 512\\3\times3, 512\end{bmatrix}$	$\begin{bmatrix}3\times3, 512\\3\times3, 512\\3\times3, 512\\3\times3, 512\end{bmatrix}$
Pool4	maxpool	maxpool
Conv5_x	$\begin{bmatrix}3\times3, 512\\3\times3, 512\\3\times3, 512\end{bmatrix}$	$\begin{bmatrix}3\times3, 512\\3\times3, 512\\3\times3, 512\\3\times3, 512\end{bmatrix}$
Pool5	maxpool	maxpool
FC1	FC-4096	FC-4096
FC2	FC-4096	FC-4096
FC3	FC-1000	FC-1000
Classifier	soft-max	soft-max

Table 2. Basic structure of ResNet.

Layer	Filter
Conv1	7x7x64, stride 2
Conv2_1	3x3 max pool, stride 2
Conv2_x	$\begin{bmatrix}1\times1,\ 64\\3\times3,\ 64\\1\times1,\ 256\end{bmatrix}\times3$ layers
Conv3_x	$\begin{bmatrix}1\times1,\ 128\\3\times3,\ 128\\1\times1,\ 512\end{bmatrix}\times4$ layers
Conv4_x	$\begin{bmatrix}1\times1,\ 256\\3\times3,\ 256\\1\times1,\ 1024\end{bmatrix}\times N$ layers
Conv5_x	$\begin{bmatrix}1\times1,\ 512\\3\times3,\ 512\\1\times1,\ 2048\end{bmatrix}\times3$ layers
Pool	average pool
FC	FC-1000
Classifier	soft-max

2.5 MobileNet

MN의 가장 큰 특징은 합성곱의 연산량을 줄여주는 depthwise separable 합성곱과 채널을 조절하는 width multiplier, 입력의 크기를 조절하는 resolution multiplier를 사용한 것이다.

Table 3은 MN1의 구조를 보여준다.[5] 여기서 conv dw는 depthwise separable 합성곱을 의미한다.

MN2의 경우 MN1에서 inverted residual 개념이 도입된 네트워크로 해당 개념은 일반적인 residual 개념인 wide-narrow-wide로 연결 생략을 합치는 것이 아닌 narrow-wide-narrow 구조로 연결 생략을 합친 구조이다.

2.6 제안하는 네트워크 구조

본 논문에서 사용한 입력의 해상도가 작아서 다른 깊은 네트워크의 구조를 그대로 이용하였을 때 깊은 층에서 입력의 크기가 1 × 1로 되는 현상이 발생하였다. 또한, 채널 수의 증가로 인한 과적합 문제가 발생하지 않게 하면서 채널의 증가함에 따른 파라미터 수의 증가를 방지하기 위해 네트워크의 최대 채널 수를 128로 제한을 두고 총 층의 깊이가 13인 네트워크를 구성하였다. 제안하는 네트워크 구조(RMN)는 Table 4에 나타나 있다.

Table 3. The structure of MN1 network.

Layer / Stride	Input Size	Filter Shape
conv / s2	224 × 224 × 3	3 × 3 × 3, 32
conv dw / s1	112 × 112 × 32	3 × 3 × 32 dw
conv / s1	112 × 112 × 32	1 × 1 × 32, 64
conv dw /s2	112 × 112 × 64	3 × 3 × 64 dw
conv1 / s1	56 × 56 × 64	1 × 1 × 64,128
conv dw / s1	56 × 56 × 128	3 × 3 × 128 dw
conv / s1	56 × 56 × 128	1 × 1 × 128, 128
conv dw / s2	56 × 56 × 128	3 × 3 × 128 dw
conv / s1	28 × 28 × 128	1 × 1 × 128, 256
conv dw / s1	28 × 28 × 256	3 × 3 × 256 dw
conv / s1	28 × 28 × 256	1 × 1 × 256, 256
conv dw / s2	28 × 28 × 256	3 × 3 × 256 dw
conv / s1	14 × 14 × 256	1 × 1 × 256, 512
conv dw / s1	14 × 14 × 512	3 × 3 × 512dw
conv / s1	14 × 14 × 512	1 × 1 × 512, 512
conv dw / s1	14 × 14 × 512	3 × 3 × 512dw
conv / s1	14 × 14 × 512	1 × 1 × 512, 512
conv dw / s1	14 × 14 × 512	3 × 3 × 512dw
conv / s1	14 × 14 × 512	1 × 1 × 512, 512
conv dw / s1	14 × 14 × 512	3 × 3 × 512dw
conv / s1	14 × 14 × 512	1 × 1 × 512, 512
conv dw / s2	14 × 14 × 512	3 × 3 × 512 dw
conv / s1	7 × 7 × 512	1 × 1 × 512, 1024
conv dw /s2	7 × 7 × 1024	3 × 3 × 1024 dw
conv / s1	7 × 7 × 1024	1 × 1 × 1024, 1204
AvgPool	7 × 7 × 1024	pool 3 × 3
FC	1 × 1 × 1024	1024 × 1000
Classifier	soft-max	

Table 4. The structure of reduced MobileNet.

Layer / Stride	Input Size	Filter
Conv1 / s2	96 × 40 × 3	10 × 4 × 3, 64
Conv2_1 / s1	48 × 20 × 64	1 × 3 × 64, 64
Conv2_2 / s1	48 × 20 × 64	3 × 1 × 64, 64
Conv2_3 / s1	48 × 20 × 64	1 × 1 × 64, 64
Conv2_4 / s1	48 × 20 × 64	1 × 3 × 64, 64
Conv2_5 / s1	48 × 20 × 64	3 × 1 × 64, 64
Conv2_6 / s1	48 × 20 × 64	1 × 1 × 64, 128
Conv3_1 / s1	48 × 20 × 128	3 × 3 × 128, 128
Conv3_2 / s1	24 × 10 × 128	1 × 3 × 128, 128
Conv3_3 / s1	24 × 10 × 128	3 × 1 × 128, 128
Conv3_4 / s1	24 × 10 × 128	1 × 3 × 128, 128
Conv3_5 / s1	24 × 10 × 128	3 × 1 × 128, 128
Conv3_6 / s1	24 × 10 × 128	1 × 1 × 128, 128
AvgPool	24 × 10 × 128	pool 3 × 3
FC	1 × 1 × 11	128 × 11
Classifier	soft-max	

III. 실험 및 결과

3.1 데이터셋

본 논문에서는 구글에서 시작 단어 검출을 위해 만든 데이터셋인 Speech Commands[10] 데이터셋을 사용하였다. 해당 데이터셋은 배경 소음과 총 35개의 단어의 발화로 구성되어 있다.

각 데이터의 파일 포맷은 PCM wav, 16 bit, 16 kHz로 구성되어 있다. 본 논문에서는 Speech Commands 데이터셋 중 파일 개수가 많고 자주 쓰는 “yes”, “no”, “up”, “down”, “left”, “right”, “on”, “off”, “stop”, “ go”를 검출하였다. 시작 단어에 대한 파일 개수는 Table 5에 나타내었다.

본 논문에서는 합성곱 신경망 방식의 시작 단어 검출을 진행하기 위해서 1차원인 PCM wav 파일을 각각 40차원의 MFCC 및 스펙트로그램 특징 추출을 통해 각각 96 × 40의 2차원 이미지로 네트워크의 입력으로 사용하였다. 여기서 96은 각 프레임을 의미하고 40은 MFCC의 경우 필터 뱅크 수를, 스펙트로그램의 경우 에너지 밴드를 의미한다.

길이가 다른 wav 파일을 동일 크기의 입력으로 사용하기 위해서 2초를 기준으로 부족한 부분은 묵음을 추가하여 입력의 크기를 조절하였다.

3.2 실험 환경

본 논문에서는 시작 단어 검출을 위해 파이썬 기반의 구글 텐서플로우[11] 딥러닝 프레임워크를 사용하였다.

본 논문에서는 총 데이터의 20 %를 검증 데이터로 사용하여 학습 과정에서 정확도가 가장 높게 나오는 네트워크의 파라미터 값을 선택하였다.

3.3 성능평가 방식

본 논문에서 제시한 네트워크들의 성능을 평가하기 위해서 분류 문제에서 많이 사용되는 오차행렬[12]를 사용하였다. 오차 행렬은 Table 6과 같다. 오차 행렬에서 TP(True Positive)는 실제값이 “A”일 때 예측값이 “A”인 경우, FN(False Negative)은 실제 값이 “B”일 때 예측값이 “A”인 경우, FP(False Positive)는 실제값이 “A”일 때 예측값이 “B”인 경우, 마지막으로 TN(True Negative)은 실제값이 “B”일 때 예측값이 “B”인 경우이다.

Table 5. Words from the Speech Commands dataset.

wake-up-word	Number of files	wake-up-word	Number of files
yes	4044	up	3723
no	3941	down	3917
left	3801	right	3778
off	3745	on	3845
stop	3872	go	3880

Table 6. Confusion matrix.

	Real “A”	Real “B”
Predict “A”	TP (True Positive)	FN (False Negative)
Predict “B”	FP (False Positive)	TN (True Negative)

오차 행렬을 통해 구해진 값들을 이용하여 각 합성곱 신경망의 재현율(Recall), 정밀도(Precision), $F_1 score$, 및 정확도(Accuracy)와 합성곱 신경망의 학습된 모델 크기를 기준으로 각 합성곱 신경망의 최종 성능을 평가한다. 재현율, 정밀도, 정확도, 및 $F_1 score$는 Eqs. (1) ~ (4)과 같다.

$$Recall = \frac{TP}{TP + FN}. \tag{1}$$

$$Precision = \frac{TP}{TP + FP}. \tag{2}$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}. \tag{3}$$

$$F_1 score = 2 * \frac{Precision * Recall}{Precision + Recall}. \tag{4}$$

3.4 실험 결과

Table 7은 스펙트로그램 특징을 이용한 시작 단어 검출의 성능, Table 8은 MFCC 특징을 이용한 시작 단어 검출의 성능을, Table 9는 각 모델의 학습된 모델 크기를 보여준다. 학습 과정에서 발생하는 오차를 줄이기 위해 네트워크마다 5번씩의 실험을 진행하

Table 7. Performance of wake-up-word detection using spectrogram feature.

Network	Accuracy	Precision	Recall	$F_1\,score$
MLP	0.302	0.972	0.970	0.971
CNN	0.791	0.997	0.987	0.992
VGG16	0.939	0.999	0.994	0.996
VGG19	0.926	0.998	0.992	0.995
ResNet50	0.939	0.999	0.998	0.999
ResNet101	0.942	0.998	0.996	0.998
ResNet152	0.935	0.999	0.997	0.998
MN1(1.0)	0.928	0.999	0.998	0.999
MN1(0.5)	0.923	0.999	0.996	0.998
MN2(1.0)	**0.943**	**0.999**	**0.996**	**0.998**
MN2(0.5)	0.940	0.999	0.998	0.999
RMN	0.921	0.999	0.997	0.998

Table 8. Performance of wake-up-word detection using MFCC feature.

Network	Accuracy	Precision	Recall	$F_1\,score$
MLP	0.655	0.994	0.987	0.990
CNN	0.770	0.994	0.943	0.967
VGG16	**0.962**	**0.999**	**0.998**	**0.998**
VGG19	0.954	0.998	0.991	0.995
ResNet50	0.949	0.999	0.998	0.998
ResNet101	0.954	0.999	0.997	0.999
ResNet152	0.952	0.999	0.995	0.997
MN1(1.0)	0.958	0.999	0.998	0.999
MN1(0.5)	0.946	0.999	0.996	0.998
MN2(1.0)	0.953	0.999	0.998	0.999
MN2(0.5)	0.950	0.999	0.998	0.999
RMN	0.951	0.999	0.997	0.999

여 평균을 구하였다.

비교에 사용된 네트워크는 MLP, CNN, VGG16, VGG19, ResNet50, ResNet101, ResNet152, MN1, MN2 및 제안 알고리즘인 RMN이다. MN1, MN2 괄호 안의 숫자는 width multiplier의 값을 의미한다.

스펙트로그램과 멜-주파수 캡스트럼 계수를 비교하였을 때 전반적으로 멜-주파수 캡스트럼 계수를 입력으로 사용하였을 때 전반적인 정확도가 높게 나타났다. 멜-주파수 캡스트럼 계수를 입력으로 하였을 때의 정밀도와 재현율은 MLP와 CNN을 제외한 대부분의 합성곱 신경망에서 0.997~0.999의 값을 나타내었다. 각 네트워크의 정확도를 비교하여 보았을 때, VGG16 네트워크의 경우 0. 962로 가장 우수한 성능을 나타내었다. 학습된 모델 크기를 보면 모바일넷을 변형한 네트워크가 1.5 MB로 가장 우수한 성능을 나타내었다.

Table 9. Trained model size using MFCC feature.

Network	Trained model size
MLP	6.2 MB
CNN	11 MB
VGG16	564 MB
VGG19	615 MB
ResNet50	269 MB
ResNet101	486 MB
ResNet152	665 MB
MN1(1.0)	37 MB
MN1(0.5)	9.5 MB
MN2(1.0)	26 MB
MN2(0.5)	116 MB
RMN	**1.5 MB**

VGG16 네트워크를 사용하였을 때의 시작 단어 인식의 정확도가 가장 우수하게 나타났지만 학습된 모델 크기가 564 MB로 높은 편에 속하였고 모바일넷을 변형한 네트워크를 사용하였을 때 정확도 0.951로 VGG16 네트워크를 사용하였을 때의 시작 단어 인식 정확도보다 0.011 낮은 성능을 나타냈다. 하지만 학습된 모델 크기는 1.5 MB로 VGG16 네트워크의 564 MB보다 1/375의 모델 크기를 나타내었다. 이는 정확도 0.958을 보이는 모바일 기기를 위한 합성곱 신경망인 MN1(1.0)의 학습된 모델 크기인 37 MB와 비교하여도 약 1/25의 모델 크기를 나타낸다.

IV. 결론 및 향후 연구 방향

본 논문에서는 모바일 기기에서 시작 단어 검출을 위하여 다양한 합성곱 신경망 방식을 이용하여 각 방식의 성능을 비교하였다. 정확도를 기준으로 하였을 때는 VGG16 네트워크가 가장 우수한 성능을 나타내지만 모바일 기기에서 시작 단어 검출을 위한 네트워크로는 0.951의 준수한 정확도, 0.999의 정밀도, 0.997의 재현율을 가지면서도 MN1(1.0) 대비 1/25

의 모델 크기를 가지는 제안 네트워크가 우수한 성능을 보이는 것을 확인하였다.

향후 모바일 기기에서의 화자 인식 알고리즘에 대한 추가연구를 진행하여 시작 단어 검출과 함께 화자를 인식할 수 있는 연구를 진행할 계획이다.

감사의 글

본 논문은 2018년도 대한민국 교육부와 한국연구재단(NRF-2018S1A5A2A03037308), 산업통상자원부의 산업기술혁신사업(10073154, 인간 내면상태의 인식 및 이를 이용한 인간친화형 인간-로봇 상호작용 기술 개발) 및 2020년도 정부의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임[2020-0-01389, 인공지능융합연구센터지원(인하대학교)].

References

1. B. H. Juang and L. R. Rabiner, “Hidden Markov models for speech recognition,” Technometrics, **33**, 251-272 (1991).
2. C. Cortes and V. Vladimir, “Support-vector networks,” Machine learning, **20**, 273-297 (1995).
3. Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proc. the IEEE. **86**, 2278-2324 (1998).
4. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” Proc. the IEEE CVPR. 1-9 (2015).
5. A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” arXiv preprint arXiv: 1704.04861 (2017).
6. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” Proc. the IEEE CVF. Conf. computer vision and pattern recognition, 4510-4520 (2018).
7. B. Logan, “Mel frequency cepstral coefficients for music modeling,” Ismir. **270**, 1-11 (2000).
8. K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556 (2014).
9. K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” Proc. the IEEE Conf. CVPR. 770-778 (2016).
10. P. Warden, “Speech commands: A dataset for limited-vocabulary speech recognition,” arXiv preprint arXiv: 1804.03209 (2018).
11. M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, “Tensorflow: A system for large-scale machine learning,” Proc. the 12th USENIX symposium on OSDI. 265-283 (2016).
12. F. Provost and R. Kohavi. “Guest editors’ introduction: On applied research in machine learning,” Machine learning, **30**, 127-132 (1998).

저자 약력

▸ 김 상 홍 (Sanghong Kim)

2018년 2월 : 인하대학교 전자공학과 학사
2020년 2월 : 인하대학교 전자공학과 석사
2020년 2월 ~ 현재 : 인하대학교 전기컴퓨터공학과 박사과정

▸ 이 보 원 (Bowon Lee)

1993년 2월 : 서울대학교 전기공학부 학사
2003년 5월 : University of Illinois at Urbana-Champaign 석사
2006년 12월 : University of Illinois at Urbana-Champaign 박사
2007년 2월 ~ 2014년 2월 : 미국 휴렛패커드 연구소(HP Labs) 연구원
2014년 3월 ~ 현재 : 인하대학교 전자공학과 교수