컨볼루션 신경망과 멜-스펙토그래픽 음성 표현을 이용한 언어 커맨드 분류

히키 크리스◦, 장병탁
서울대학교
{chickey, btzhang}@bi.snu.ac.kr

Classifying verbal commands using Convolutional Neural Networks and Mel-spectrographic sound representations

Chris Hickey, Byoung-Tak Zhang
Seoul National Univeristy

요 약

Consumer products are increasingly becoming dependent on Keyword Spotting (KWS) in order to work effectively. The present study aims to explore whether image classification techniques could be used to provide accurate and robust keyword recognition systems on a fixed predefined set of words taken from the Google Speech Command dataset. Additionally, the present study aims to establish which visual representation of audio is most suitable for CNN classification. After testing four different visual representations of the audio data, mel-scale decibel spectrograms were found to be the most suitable representation for learning through Convolutional Neural Networks. Convolutional Neural Networks were able to achieve near state-of-the-art performance at 89.5% on the test dataset for this KWS classification task. The present study shows that Convolutional Neural Networks may be able to rival other common models such as RNN-LSTM in recognising keyword commands. Future study should investigate CNN keyword classification performance on larger vocabulary sets.

1. Introduction

In recent years, command based consumer technology like Amazon Echo, Siri, and Google Now have exploded in popularity. As such, the ability of technology to accurately identify and decipher speech commands is increasingly important. As Keyword Spotting (KWS) technology continues to improve over time, such technology could provide a completely hands-free interface to electronics. This would not only be more convenient than typing by hand, but it may also facilitate a safer interface with technology when driving or during an emergency.

Convolutional Neural Networks (CNN) architectures such as AlexNet[1] and VGG[2] have proven extremely successful for classifying extremely large image sets. Recent research by scientists at Google[3] has applied a similar architecture to the domain of audio classification. Their research successfully used large scale CNN architectures to solve Acoustic Event Detection classification tasks on the YouTube-100M dataset. This network was capable of identifying acoustic audio events such as the sounds of specific instruments (trumpet, guitar etc.) with high accuracy.

The aforementioned study by Google visualized sound to the extent that it was suitable for input to CNN by using mel spectrograms. Mel spectrograms are representations through a scale of pitches judged by humans to be equal in distance. Given that the mel scale is a representation of sound specifically designed to encapsulate differences in pitch as determined by human listeners, this visual representation of sound may be uniquely suited to the KWS problem. However, various alternative waveform and spectrographic representations are possible for this task. Such representations which have been used in previous studies include Hertz frequency spectrographic representations[4], Short-time Fourier transform (STFT) spectrographic representations, and raw audio wave signal representations[5].

2. Dataset

This paper utilizes the Google Speech Command Dataset[6]. Originally released by Google in August 2017, this dataset includes 65,000 speech audio samples, taken from thousands of individuals. Each audio clip is an approximately 1 second long mono WAV file. The dataset contains 30 command words: *bed, bird, cat, dog, down, eight, five, four, go, happy, house, left, marvin, nine, no, off, on, one, right, seven, sheila, six, stop, three, tree, two, up, wow, yes, zero*. There are over 1,000 samples for each of the 30 commands, and background noises have been added to the command samples to make the dataset more difficult to learn. Dataset sampling rate is 16KHz, and sampling resolution is 16bit. Current state-of -the-art performance on this dataset is approximately 90.5% according to the dataset leaderboard on Kaggle[7].

3. Model

This paper uses a similar CNN architecture to Choi and colleagues'[8] recent CNN, used for music classification. The CNN architecture used in this paper had four convolutional layers. Max pooling, relu activation and batch normalization were applied

after the first convolutional layer. Max pooling, elu activation and dropout of 0.5 were applied after each subsequent convolutional layer. Flattening was then used to reshape the input into a vector, which was then passed onto a fully connected layer. Dropout of 0.6 was then applied, followed by softmax for the output activation layer.

4. Preprocessing

All audio files were resampled to 44100Hz. Four different spectrogram representations of each audio file in the dataset were extracted: *STFT amplitude spectrograms, STFT decibel spectrograms, mel-scale amplitude spectrograms, and mel-scale decibel spectrograms*. Padding was added so that all audio files were the same length. All audio preprocessing was done using the librosa python library[9].

5. Experiment

Four separate models were first trained for 20 epochs, each using an adadelta optimizer[1]. Each of these models used a different spectrogram representation of the audio files as input. Further training was then carried out for an additional 80 epochs using the audio representation which performed best during the initial experimentation phase. Mini-Batches of size 32 were used during both phases of training. Data was divided using roughly a 70%/15%/15% Training/Validation/Test split.

6. Results

In the initial short experimentation phase, all four models achieved over 80% accuracy on both the training and test datasets after 20 epochs. Note that in Table 1, the "Train %" column refers to the combination of both the Training and Validation datasets that were used during model training. The mel-scale decibel spectrogram representations (see Figure 1) had the highest accuracy after 20 epochs. Therefore training was continued using only the model trained with these spectrograms.

Spectrogram	Train %	Test %
Mel-scale Decibel	85.2	84.3
Mel-scale Amplitude	83.9	83.4
STFT Decibel	82.1	81.5
STFT Amplitude	82.1	81.9

표 1: Accuracy for each of the four models trained with different spectrograms after 20 epochs.

After 100 epochs, the model trained using mel-scale decibel spectrograms successfully achieved 90% and 89.5% accuracy on both the training and test datasets respectively. These results are very similar to the current state-of-the-art model performance on this dataset of 90.5%

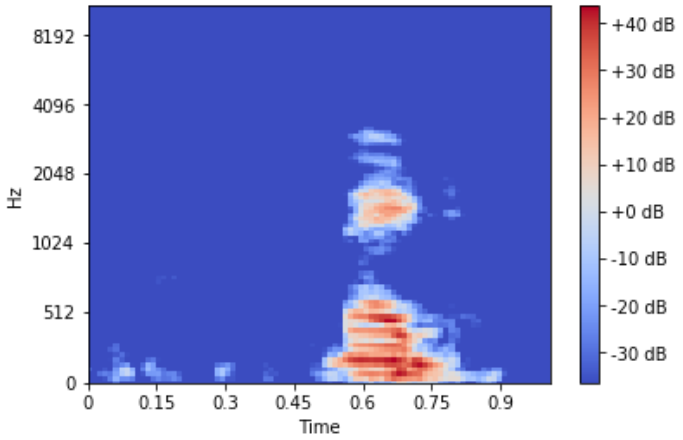

그림 1: Sample of audio file representation used as input for CNN. This figure gives a sample mel-scale decibel spectrogram of a sample of the word bed taken from the Google dataset.

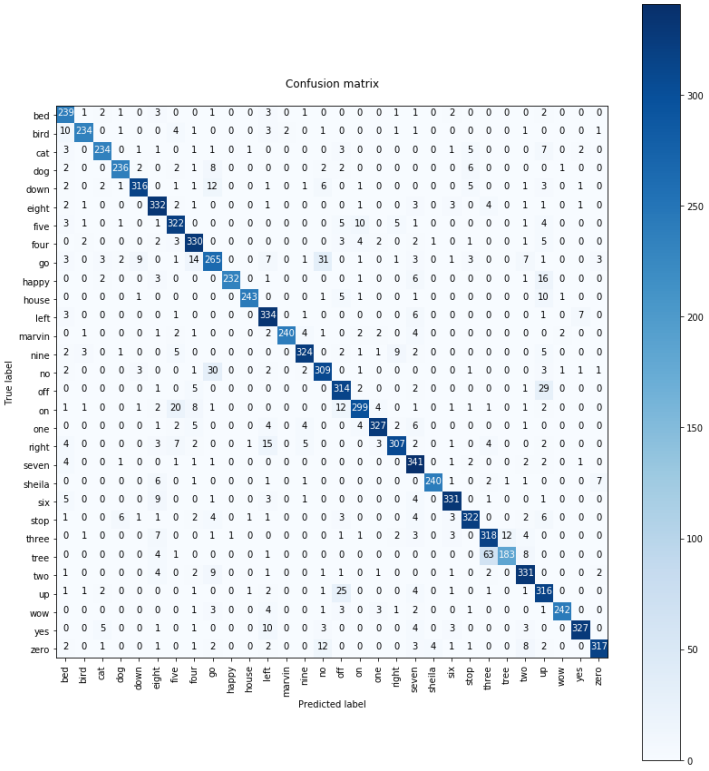

그림 2: Confusion matrix of true class values and predicted class values for the test dataset.

When looking at the confusion matrix (see Figure 2) of results for the test set using the model trained on mel-scale decibel spec-

1) Code for running these experiments can be found at `https://github.com/chrishickey/CNNCommandClassifier`

trograms, some interesting observations can be made. Firstly the most commonly misclassified words were tree and three. This makes sense as these two words are extremely similar. Other commonly misclassified words were go and no. Again, since these are two similar sounding, single syllable words, one could reasonably imagine a human listener also misclassifying these words. As such this confusion matrix appears to suggest that the misclassifications produced by the modal are similar to misclassifications that a human listener may make.

7. Conclusion

The contributions of this paper are as follows:

- Mel-scale decibel spectrogram representations of audio data are the most suitable visual representations of keyword speech for training Convolutional Neural Networks.
- Convolutional Neural Networks can be used to provide state-of-the-art performance on KWS classification tasks with a small fixed number of keywords.

This current research was performed on a very small sample size of keywords. Similarly, RNN-LSTM[10] architectures have been used to solve KWS problems both on the Google Speech Command dataset and others. Future research should compare how these two architectures perform as the number of words in the KWS classification challenge increases. While both architecture types exhibit excellent performance on datasets with a small number of words, future research should investigate which architecture performs best when the vocabulary of possible keywords for classification expands from 30 to 300 or even 3,000. Moreover, if performance degenerates at scale for one type of network architecture but not the other, investigating the reasons as to why one architecture is outperforming the other would be an interesting potentially beneficial avenue of research to explore.

8. Acknowledgements

This work was partly supported by the Institute for Information & Communications Technology Promotion (2015-0-00310-SW. StarLab, 2017-0-01772-VTT, 2018-0-00622-RMI, 2019-0-01367-BabyMind) and Korea Institute for Advancement Technology (P0006720-GENKO) grant funded by the Korea government.

참고 문헌

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

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

[3] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, *et al.*, "Cnn architectures for large-scale audio classification," in *2017 ieee international conference on acoustics, speech and signal processing (icassp)*, pp. 131–135, IEEE, 2017.

[4] O. Janssens, V. Slavkovikj, B. Vervisch, K. Stockman, M. Loccufier, S. Verstockt, R. Van de Walle, and S. Van Hoecke, "Convolutional neural network based fault detection for rotating machinery," *Journal of Sound and Vibration*, vol. 377, pp. 331–345, 2016.

[5] S. Qu, J. Li, W. Dai, and S. Das, "Understanding audio pattern using convolutional neural network from raw waveforms," *arXiv preprint arXiv:1611.09524*, 2016.

[6] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[7] "Tensorflow speech recognition challenge." `https://www.kaggle.com/c/tensorflow-speech-recognition-challenge/leaderboard`.

[8] K. Choi, G. Fazekas, and M. Sandler, "Automatic tagging using deep convolutional neural networks," *arXiv preprint arXiv:1606.00298*, 2016.

[9] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, vol. 8, 2015.

[10] P. Phan, T. M. Giang, L. Nam, *et al.*, "Vietnamese speech command recognition using recurrent neural networks," *IJACSA) International Journal of Advanced Computer Science and Applications*, vol. 10, no. 7, 2019.