Seminar 2

Keyword Spotting System with Speech Command Dataset

Trinh Tuan Anh

Supervisor: **PhD. Trần Thị Thảo & PhD. Trần Thị Anh Xuân**

22/05/2021

Mục lục

Agenda

Introduction

Keyword Spotting definition

Keyword Spotting - process of identifying pre-defined keywords, in speech recorded in real-time.

Keyword Spotting in real world

Wakeup Word - common way to begin an interaction by the voice interface.

Example

- Amazon - "Alexa"
- Google - "OK Google"
- Apple - "Hey Siri"

Fig 1. Keyword spotting on production.

Agenda

Theoretical basis

Convolution

Convolution provides a way of 'multiplying together' two arrays of numbers, generally of different sizes, but of the same dimensionality, to produce a third array of numbers of the same dimensionality[1].

The convolution is performed by sliding the kernel over the image, generally starting at the top left corner, so as to move the kernel through all the positions where the kernel fits entirely within the boundaries of the image. Each kernel position corresponds to a single output pixel, the value of which is calculated by multiplying together the kernel value and the underlying image pixel value for each of the cells in the kernel, and then adding all these numbers together.

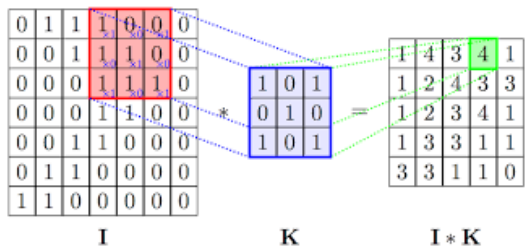

Fig 2. Convolution

Theoretical basis

1D Convolution

1D Convolution is a sort of convolution which is for signals or speech.

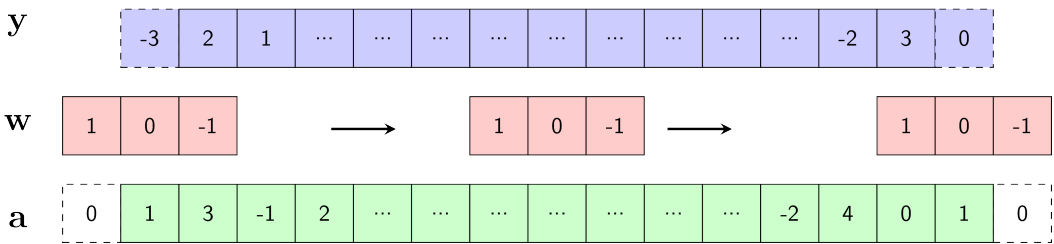

Fig 3. 1D Convolution

Theoretical basis

Pooling layer

Pooling is the process of extracting the features from the image output of a convolution layer. This will also follow the same process of sliding over the image with a specified pool size/kernel size.

There are two types of pooling is available:

1. Max Pooling
2. Average Pooling

Max Pooling is being used widely and it will just keep the highest number in the pool and discard the rest. By getting the highest value in each pool we will be getting the significant features of the image, the lower values are not the features at all or not significant features to be able to use in the model.

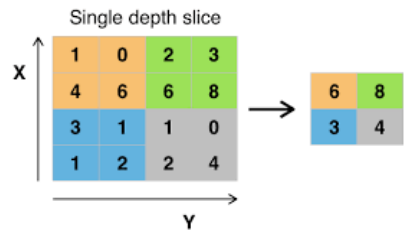

Fig 4. Example of Max Pooling

Theoretical basis

Batch Normalizing Transform

Batch normalization (or batch norm) is a method used to make NNnet faster and more stable through normalization of the layer's inputs by re-centrering and re-scaling.
Use B to denote a mini-batch of size m of the entire training set. The empirical **mean** and **variance** of B could thus be denoted as:

$$\mu = \frac{1}{m}\sum_{i=1}^{m} x_i \; ; \sigma^2 = \frac{1}{m}\sigma_{i=1}^{m}(x_i - \mu_B)^2 \tag{1}$$

For a layer of the network with d-dimensional input $x = (x^{(1)}, ..., x^{(d)})$, each dimension of its input is then normalized (i.e. recentered and re-scaled) separately.

$$\hat{x}^{(k)} = \frac{x_i^{(k)} - \mu_B^{(k)}}{\sqrt{\sigma_b^{(k)^2} + \epsilon}} \tag{2}$$

where $k \in [1, d]$; $i \in [1, m]$ and $\mu_B^{(k)}$, $\sigma_B^{(k)^2}$ are the per-dimension mean and variance, respectively.

Theoretical basis

Softmax function

The softmax function is a generalization of the logistic function to multiple dimensions. It is used in multinomial logistic regression and is often used as the last activation function of a neural network to normalize the output of a network to a probability distribution over predicted output classes.

The softmax function takes as input a vector z of K real numbers, and normalizes it into a probability distribution consisting of K probabilities proportional to the exponentials of the input numbers.

The standard (unit) softmax function $\sigma : \mathbb{R}^K \rightarrow [0,1]^K$ is defined by the formula:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \text{ for } i = 1, ..., K \text{ and } z = (z_1, ..., z_K) \in \mathbb{R}^K \tag{3}$$

Agenda

Problem analysis

Objective

Detect whether the input speech is one of 35 keywords or not.

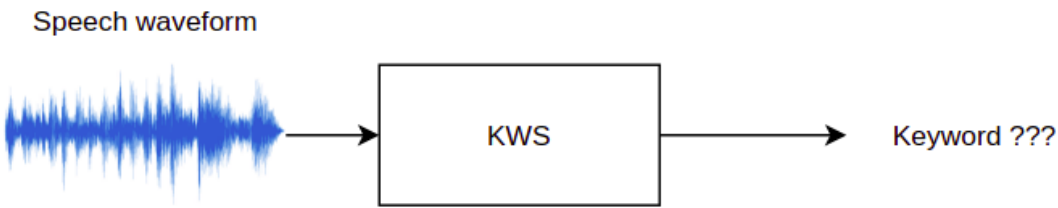

Fig 5. Keyword Spotting System structure

Problem analysis

Model was used:

- Based on M5 model [2]
- The struture of M5 is described in figure 6

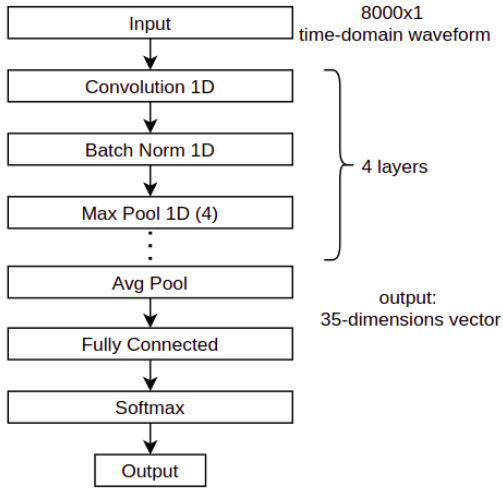

Fig 6. M5 structure

Problem analysis

Dataset

- **Speech Commands** dataset of Cornell University
- Number of keywords: 35
- Number of sample file: 105,835
- Sample rate: 16000 Hz

Data division

- Size of train data: 105,829 files
- Size of test data: 11,005 files

Word	Number of Utterances
Backward	1,664
Bed	2,014
Bird	2,064
Cat	2,031
Dog	2,128
Down	3,917
Eight	3,787
Five	4,052
Follow	1,579
Forward	1,557
Four	3,728
Go	3,880
Happy	2,054
House	2,113
Learn	1,575
Left	3,801
Marvin	2,100
Nine	3,934
No	3,941
Off	3,745
On	3,845
One	3,890
Right	3,778
Seven	3,998
Sheila	2,022
Six	3,860
Stop	3,872
Three	3,727
Tree	1,759
Two	3,880
Up	3,723
Visual	1,592
Wow	2,123
Yes	4,044
Zero	4,052

Agenda

Experience Result

Loss function:

- Optimizer: torch.optim.Adam
- Loss after 10 epochs (4140 batch) in figure 7

Accurancy

- Accurancy $\approx$ 80%

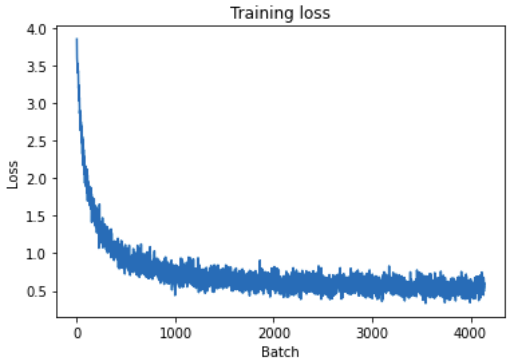

Fig 7. Training loss

Experience Result

Compare with other models in article

Model	Accurancy
M3	56.12%
M5	63.43%
M11	69.07%
M18	71.68%
M34	63.47%

Table 1. Accurancy of other model (Dataset: UrbanSound8k)

Reference


T. Vu-Huu, [Online]. Available: https://machinelearningcoban.com/2018/10/03/conv2d.


W. Dai, C. Dai, S. Qu, J. Li, and S. Das, "Very deep convolutional neural networks for raw waveforms," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 421–425. DOI: 10.1109/ICASSP.2017.7952190.

Thank you for listening!