DEVELOPMENT OF MODELS FOR SPEECH RECOGNITION AND NATURAL LANGUAGE UNDERSTANDING USING IOT MODULES WITH PARALLEL ARCHITECTURES

Snezhana Pleshkova, Aleksander Bekyarski

Snezana Pleshkova, Technical University of Sofia, Faculty of Telecommunication, Sofia 1000 8, Kl. Ohridski blvd,
E-mail: snegpl@tu-sofia.bg

Aleksander Bekyarski, Technical University of Sofia, Faculty of Telecommunication,
Sofia 1000 8, Kl. Ohridski blvd,
E-mail: aabbv@tu-sofia.bg

Abstract

Speech recognition is one of the main methods by which artificial intelligence models a person's ability to perceive and communicate through speech. In order to achieve in speech recognition the human ability to perceive and understand speech, it is necessary to improve the already existing and achieved in practical use methods and algorithms for speech recognition and natural languages understanding. This can be done by creating pre-designed models with established accuracy to be used in various specific practical implementations of speech recognition applications. The purpose of this article is to use such predefined models and embed them in modules of Internet of Things (IoT), which have a parallel architecture and would allow real-time speech recognition.

Keywords – Speech recognition models, Parallel IoT architecture, Natural languages understanding.

1. INTRODUCTION

There are many years and many researches in area of speech recognition, which are founded the principles and the basis of the methods and algorithms for speech recognition [1]. But the real practical usage of these, developed recently methods and algorithms, is realized now with the wide spread applications of artificial intelligence in many areas of human live [2]. The aims to modelling in artificial intelligence the human speech perception and understanding ability can be satisfied using the existing and improving speech recognition methods and algorithms. One of the promising way for improving the speech recognition is to prepare the predefined speech recognition models, using them in different speech recognition applications and combining them with the achievements in area of natural languages understanding [3]. Therefore, the goal of this article is to apply some of the developed predefined speech recognition models and embedded them in modules of internet of things (IoT). It is proposed also in this article to apply internet of things (IoT) modules with parallel architecture to achieve the real time work of speech recognition and natural language understanding embedding in them the proposed and developed speech recognition models. This proposition is in accordance that the internet of things (IoT) modules with parallel architecture are software compatible with almost of the existing predefined speech recognition models. Therefore, in the next section of this article are presented and described in details the developed speech recognition models, the choice of the internet of things (IoT) modules with parallel architecture for embedding in them the developed speech recognition models and natural language understanding.

2. DEVELOPMENT OF MODELS FOR SPEECH RECOGNITION SUITABLE TO IMPLEMENT IN IOT MODULES

There is a lot of predefined models for speech recognition [4], from which can to develop the desired model, according to concrete specifications of each speech recognition application. In general sense speech recognition models are the important parts necessary to exist in each block schemas and algorithms for speech recognition, as it is shown in Fig. 1.

The speech recognition models in Fig.1 are defined as separated acoustic, language and speech processing models, but they are intended to work together in algorithms for speech recognition. In acoustic model are included the specific characteristics of speech, i.e. the acoustic model contain the typical speech features, according to human

speech production, The language model include the necessary linguistic features of the specific natural language for which speech recognition is performed. Both, defined as acoustic and language models are necessary and are combined for using in speech processing model. As main input of this model are used usually the preliminary calculated speech features from the speech source. On the additional input to the speech processing model are submitted the data from acoustic and language models. If the speech recognition is performed only for the recognition of isolated words, the language modes is not obligatory, but for recognition the sentences of speech the language modes must exist. It can therefore be argued that the recognition process is the most important and therefore it is the subject of development in this article.

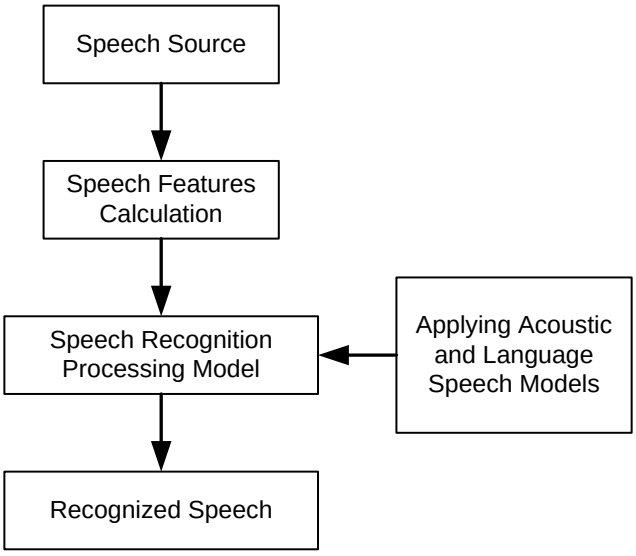

Fig.1. General view of typical block schema of speech recognition

On Fig. 2 is presented the proposed speech recognition processing model, based on the presented in Fig. 1 general view of typical block schema of speech recognition. The schema begin with speech source and speech features calculation chosen as Mel Frequency Cepstral Coefficients (MFCCs) [5]. The blocks of proposed speech recognition processing model blocks, Block 0, Block 1, Block 2 ….. Block N and Block N+1, are surrounded by a dotted line in Fig. 2. Block 0 is the first block in speech processing model. It serves as a liaison with the Mel Features Calculation block, from which are input to the speech processing model the calculated MFCCs speech model, from which features. Block N+1 is the last block in speech processing model. From this Block N+1 are output the results of recognized speech to the corresponding block, as it is shown in Fig. 2. The remaining blocks Block 1, Block 2 ….. Block N are the main part of the proposed speech recognition processing model. All of the blocks, Block 0, Block 1, Block 2 ….. Block N and Block N+1, included the similar actions as 1D Convolution, Batch Normalization, Rectified Linear Unit (ReLU), but their size and other characteristics are or can be chosen different for each of the blocks. The kernels in 1D Convolution are with *k* length, which is with different values for each of the blocks. The sequence of blocks 0, 1, 2 … N, N+1 represent the layers of deep learning neural network. The activation function of each layer of neural network is chosen to be realized as Rectified Linear Unit (ReLU). It is added in each of the blocks 0, 1, 2 … N, N+1 the Batch Normalization function to improve the learning speed of neural network and to provide regularization, avoiding overfitting. Each of the blocks 1, 2 …… N include L Sub blocks. The number of output channels of all sub blocks in each of the blocks 1, 2 …… N are equal. The input and output of each of the blocks 1, 2 …… N are connected via residual connection including the summing unit, as it is shown in Fig. 2.

The architecture of the sequence of sub blocks in each of the blocks 1, 2 …… N is the same and in presented in Fig. 3 only for a given block *i* from all blocks 1, 2 …… N. It is seen from Fig. 3, that the input and output of the block *i* are connected to the corresponding output and input of the block *i-1* and *i+1*. Each sub block include the same actions 1D Convolution, Batch Normalization, Rectified Linear Unit (ReLU), as it is presented for each of the blocks 1, 2 …… N in Fig. 2. On the detailed presentation in Fig. 3, at the output of each sub block, after Rectified Linear Unit (ReLU), is included the action dropout. The dropout operation is used to prevent overfitting in deep neural networks at the stage of learning. Also in Fig. 3 is presented more precise the residual connection, shown briefly in Fig. 2. It is seen from Fig. 3 that the residual connection is directly between input of block *i* and the last sub block, but in residual connection are included 1D Convolution and Batch Normalization. Also the Rectified Linear Unit (ReLU) and Dropout in the last sub block are after summing unit.

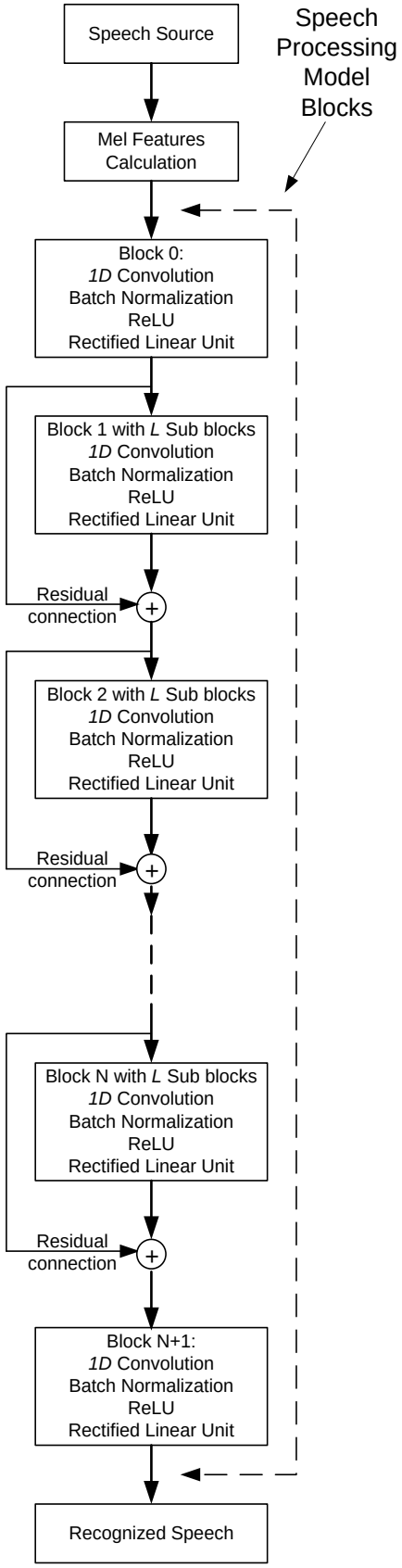

Fig. 2. Speech recognition processing model

The described above residual connection with included 1D Convolution is used in deep learning neural networks to take in account the different number of input and output channels and then pass the result through Batch Normalization.

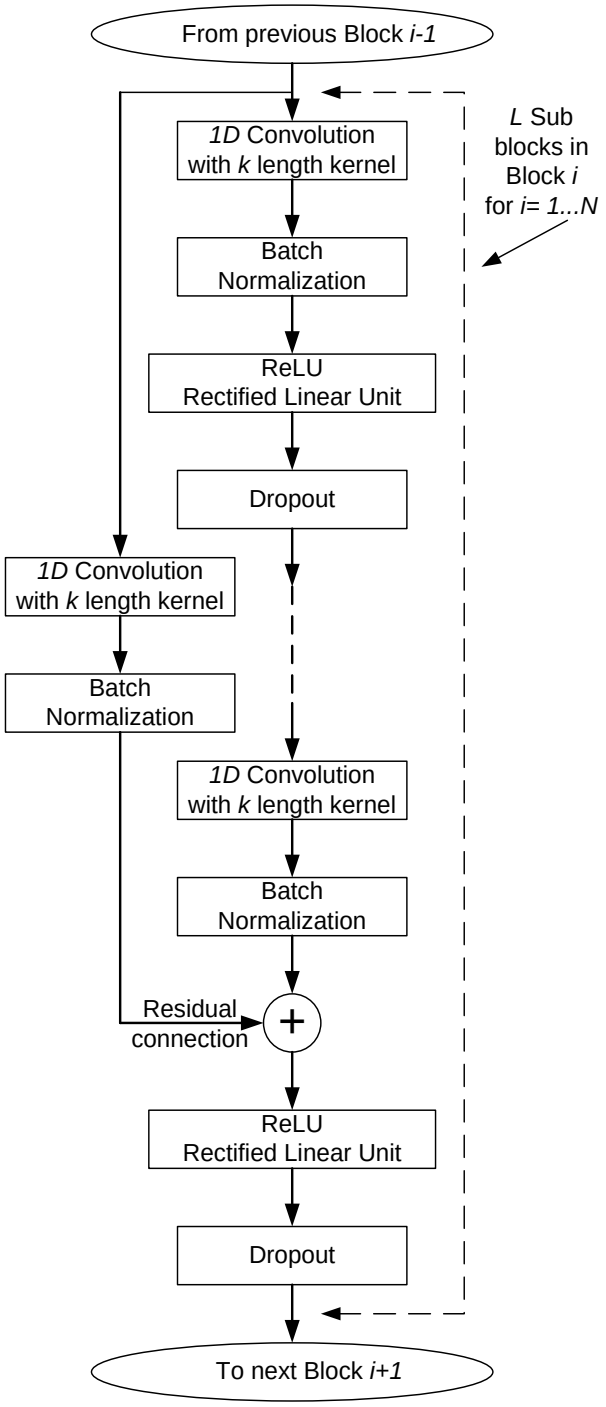

Fig. 3. Detailed schema presentation of the sub blocks in each of the blocks 1, 2 …… N

3. IMPLEMENTATION OF THE PROPOSED SPEECH RECOGNITION MODEL TO BE EMBEDDED IN IOT MODULE WITH PARALLEL ARCHITECTURE

The proposed and described above speech recognition model is tested as embedded in internet of things (IoT) module. It is necessary to satisfy the requirements of real time speech recognition, especially at applications such as natural language understanding. Therefore, it is proposed to embed

the described above speech recognition model in IoT module with parallel architecture. One of the most frequently used IoT modules in the implementation of neural networks with deep learning are IoT modules of NVIDIA [6]. In this article is proposed to use Jetson Nano IoT module shown in Fig. 4 with usually connected Keyboard, HDMI Monitor and necessary for recognizing input speech and for listen the input or recognized output speech USB Microphone and USB Speaker, respectively. In additional it is included the Internet connection to use the existing in Clouds Speech Models and Data Base of Natural Languages Understanding (NLU).

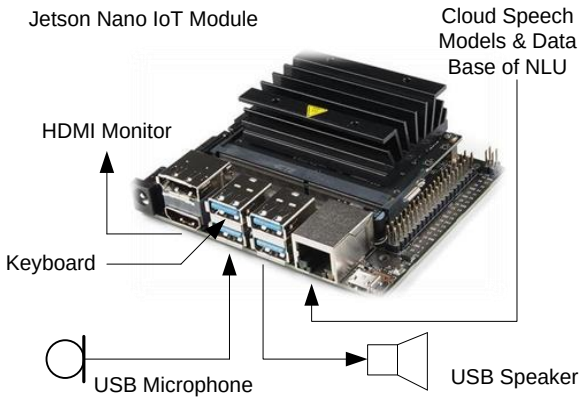

Fig. 4. Jetson Nano IoT module using to embed the proposed speech recognition model

The presented in Fig. 4 configuration of IoT module Jetson Nano is used as the base to develop the code and to test experimentally a concrete example as program application with implementation of proposed speech recognition model. For brevity, only the following important program modules of this developed as Python code are presented below to outline the main parts of proposed speech Load files to train or to test speech recognition model, presented in Fig. 2 and Fig. 3.

```
# Speech Source Module
# Load files to train or to test speech recognition
import librosa
import IPython.display as ipd
example_file = data_dir + '/Train1.wav'
audio, sample_rate = librosa.load(example_file)
ipd.Audio(example_file, rate=sample_rate)

# Mell Features Calculation of
# loaded files to train or to test speech recognition
import numpy as np
spec = np.abs(librosa.stft(audio))
spec_db = librosa.amplitude_to_db(spec, ref=np.max)

# Train the proposed speech recognition model
# using the loaded train files
import pytorch_lightning as pl
trainer = pl.Trainer(gpus=1, max_epochs=50)
trainer.fit(speech_model)

# Test the proposed speech recognition model
# using the loaded test files
import pytorch_lightning as pl
test = pl.Test(gpus=1, max_epochs=50)
test.fit(speech_model)
```

4. EXPERIMENTAL RESULTS

It can be outlined the following achieved results from experimentally tests of the proposed speech recognition model prepared as concrete example realized in program application with implementation in IoT module Jetson Nano, presented on Fig. 4. The main characteristic of the experimental speech recognition model are listed in Table 1.

TABLE I
CHARACTERISTICS OF SPEECH RECOGNITION MODEL

Number of blocks	10
Number of sub blocks in each block	5
Kernels of 1D Convolution	From 11 to 25 in Blocks 1 to 10
Output channels	From 256 to 768 in Blocks 1 to 10
Dropout	0.3

In the carried out tests are prepared the comparison with the existing in Cloud Speech Models and Data Base for Natural Language Understanding [7]. The comparison of results for speech recognition, using the developed and existing models using the same

examples of speech samples (as isolated words and sentences) and data base of natural language model are presented in Table 2.

TABLE II
COMPARISON OF THE DEVELOPED SPEECH RECOGNITION MODEL WITH SOME EXISTING SPEECH RECOGNITION MODELS

Existing Models [7]	Comparison of precision in % of speech recognition using the developed and the existing models
Model 1	82% to 89%
Model 2	76% to 88%
Model 3	83% to 97%
Model 4	87% to 95%

5. CONCLUSION

The briefly presented in Table 2 overall comparative results for speech recognition, using the developed and existing models can be comment in the following way.

In general the results of speech recognition, using the proposed speech recognition model, are similar (but with lower 76-87 % accuracy) in comparison of the same examples of speech recognition using the existing models (88-97%). This can be explain with the following arguments:

- insufficient in-depth training of the proposed model compared to the existing ones;
- more precisely defining the parameters of the neural network with deep training for the existing speech recognition models;
- greater number of blocks, corresponding sub blocks and the number of channels in the layers of neural network with deep learning in existing speech recognition models, using in comparison.

Regardless of this differences from the existing speech recognition models, the following advantages of the proposed speech recognition model can be highlighted:

- simpler scheme suitable for embedding in IoT modules with less complexity of parallel architecture, like IoT module Jetson Nano;
- preference for use in simple practical applications for speech recognizing a limited number of words and sentences and from a specific natural language for examples mobile robots or other devices, using limited words and sentences as speech commands in voice control.

ACKNOWLEDGMENTS

The authors would like to thank the Research and Development Sector at the Technical University of Sofia for the financial support.

References

[1] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," Neural Networks, vol. 18, pp. 602–610, 2005.

[2] By Janna Anderson, Lee Rainie and Alex Luchsinger, "Artificial Intelligence and the Future of Humans", Pew Research Center, December, 2018.

[3] P. Liang, "Natural Language Understanding: Foundations and State-of-the-Art", ICML, July 6, 2015.

[4] D. Wang, X. Wang and S. Lv, "An Overview of End-to-End Automatic Speech Recognition", Symmetry 2019, 11.

[5] H. Perez-Meana and E. Escamilla-Hernandez, "Speaker recognition using Mel Frequency Cepstral Coefficients (MFCC) and Vector quantization (VQ) techniques", CONIELECOMP, 22nd International Conference on Electrical Communications and Computers, 2012.

[6] Jetson Nano Developer Kit, https://developer.nvidia.com/jetson-nano-developer-kit

[7] NVIDIA Cloud Computing. https://www.nvidia.com/en-us/data-center/gpu-cloud-computing