

Emergency Sound Detection – *WiSARD*

CPS760 – Internet da Coisas – Claudio Miceli

Luiz Marcio Faria de Aquino Viana, M.Sc.

E-mail: Imarcio@cos.ufrj.br

Imarcio@tlmv.com.br

luiz.marcio.viana@globo.com

luiz.marcio.viana@gmail.com

Phone: +55-21-99983-7207

DRE: 120048833

CPF: 024.723.347-10

RG: 08855128-8 IFP-RJ

Conteudo

- 1. Objetivo do Estudo**
- 2. Detecção de Sons de Emergencia**
 - 2.1. Definição do Problema**
 - 2.2. Federated Learning (FL)**
 - 2.2. TinyML**
- 3. Solução Usando *WiSARD* (RNSP)**
 - 3.1. Implementação Própria da *WiSARD***
- 4. Analise dos Experimentos**
- 5. Conclusao**

Objetivo do Estudo

#1. O objetivo deste trabalho é implementar uma aplicacao de deteccao de sons de emergencia, usando *smartphones* e *cloud* computing, e estudar os conceitos de *Federated Learning (FL)* e *TinyML*, que sao necessarios nas implementacoes em microcontroladores com pouca capacidade de processamento.

#2. Neste trabalho implementamos um algoritmo de redes neurais sem peso, similar ao *WiSARD*, para deteção dos sons de emergencia.

Deteção de Sons de Emergencia

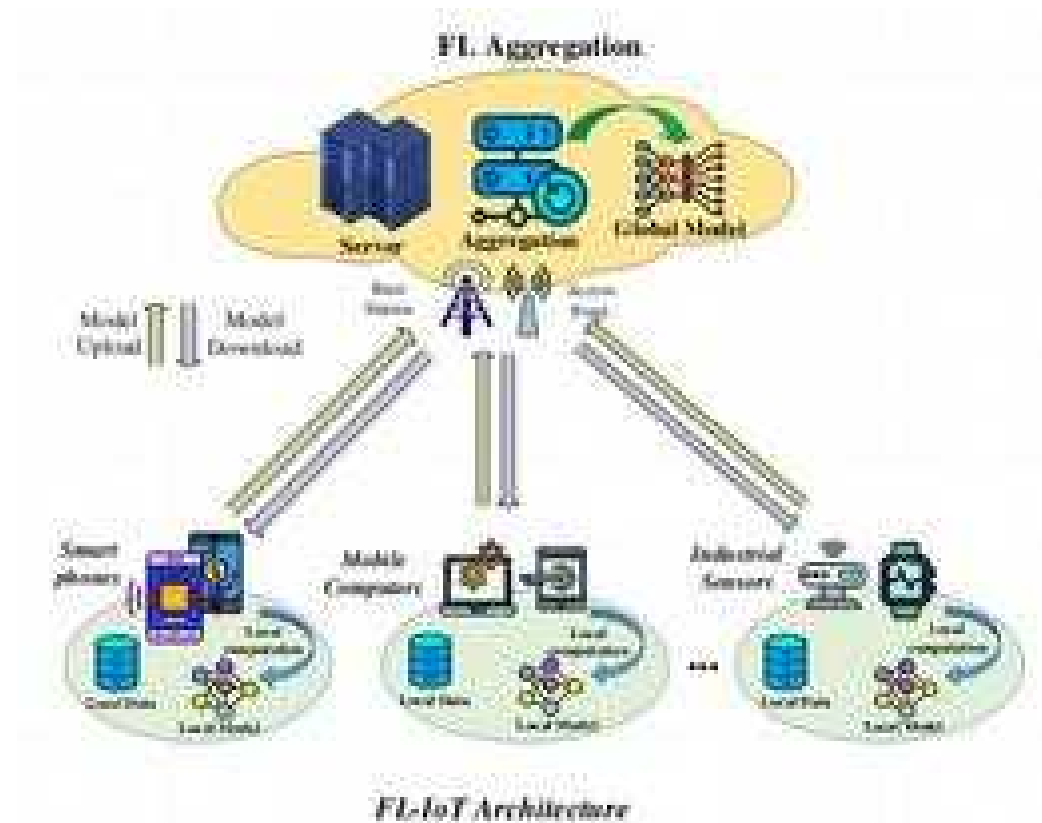
MOTIVAÇÃO:

Identificação de sons de emergencia é um processo fundamental na análise automatizada de audio.

UrbanSound8K:

Com o objetivo de melhorar os processos de identificação de sons de tiros de armas de fogo, usamos o *Dataset* com sons urbanos do Kaggle, denominado UrbanSound8K.

Federated Learning (FL)



TinyML

ATMega328

- . Low Power 8-Bit Microcontroller
- . Advanced RISC Architecture
 - 32 x 8 General Purpose Working Registers
 - Up to 20 MIPS Throughput at 20 MHz
 - On-chip 2-cycle Multiplier
- . High Endurance Non-volatile Memory Segments
 - 4/8/16/32K Bytes of In-System Self-Programmable Flash program memory
 - 256/512/512/1K Bytes EEPROM
 - 512/1K/1K/2K Bytes Internal SRAM
- . Price: US\$ 1.11/unit to US\$ 1.21/unit
- . Reference: Alibaba.com in 4th September, 2022.

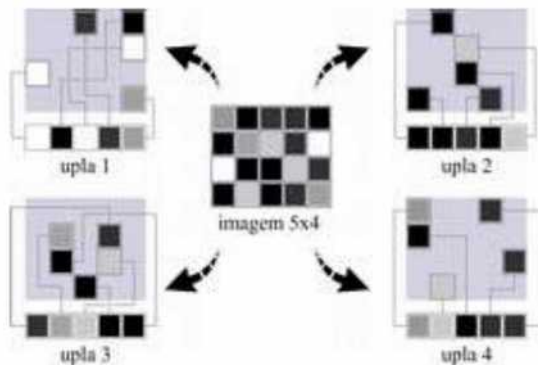
Microchip PIC16F873A

- . Low Power 8-Bit Microcontroller
- . Instruction Set
 - 35 Instructions
- . Operating Frequency - DC – 20 MHz
- . Flash Program Memory (14-bit words) – 4 KB
- . Data Memory - 192 bytes
- . EEPROM Data Memory - 128 bytes
- . Price: US\$ 2.00/unit to US\$ 5.00/unit
- . Reference: Alibaba.com in 4th September, 2022.

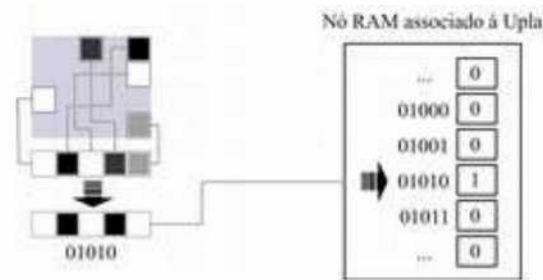


Solução Usando *WiSARD* (RNSP)

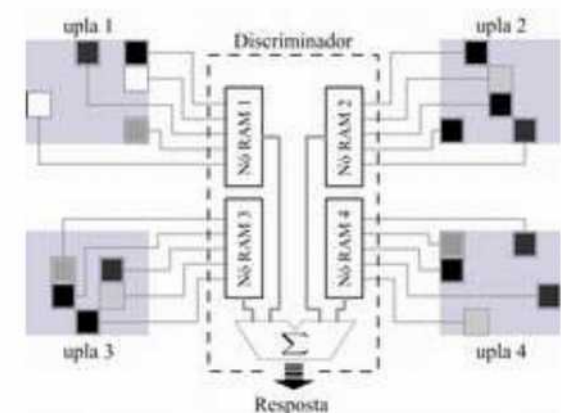
WiSARD: Para usar a rede neural sem peso *WiSARD*, a imagem é particionada em conjuntos de pixels de dimensões fixas, e denominado de n-upla. Essa particao e formada aleatoriamente, mas se mantém fixa por todo o ciclo de vida da rede neural. Cada conjunto de n-upla de pixels é associada a um nó de memória, onde cada endereço aponta para um valor de 1 bit apenas. Todas as n-uplas da imagem possuem o seu respectivo nó de memória, sendo que cada um desses nós possui um espaço de endereçamento exclusivo.



Exemplo de particionamento dos pixels da imagem em conjuntos de n-upla.



Exemplo de associação entre um conjunto de n-upla e um nó de memória.



Exemplo de espaço de memória usado para armazenar as ocorrências de cada n-upla.

Solução Usando *WiSARD* (RNSP)

TREINAMENTO: Um limiar é aplicado a cada n -upla de pixels convertendo essa n -upla em um vetor binário de n -upla bits. Esse vetor por sua vez é usado como um endereço para o nó RAM associado a essa n -upla. O valor de um dígito binário para o qual esse endereço aponta é ajustado para "1". Esse processo é executado em cada nó RAM pertencente ao neurônio que corresponde à classe a ser treinada.

CLASSIFICAÇÃO: Endereços para nós RAM são extraídos das diversas n -uplas de pixels que compõem a partição da imagem da mesma maneira como é feito no caso do treinamento. Um contador em cada neurônio é zerado no início da operação de classificação de uma imagem e, a cada n -upla onde o endereço gerado aponta para um "1", este contador é incrementado. O valor final desse contador constitui a resposta do neurônio à imagem apresentada. Vence o neurônio que apresentar a resposta mais alta, ou seja, o neurônio cujo contador tiver a soma mais alta.

Implementação Própria da *WiSARD*

#1. A implementação da rede neural sem peso *WiSARD*, é realizada usando um mapa de memória, identificado pela classe *AudioPattern*, da aplicação que foi desenvolvida para este estudo.

#2. Os sons de entrada são pré-processadas para reduzir o sua amplitude para valores de 0 a 256, em seguida são separadas em 32 faixas de valores.

#3. A rede neural sem peso *WiSARD* é usada na fase identificação do som como som de tiro de arma de fogo.

Analise dos Experimentos

#1. Os experimentos deste estudo foram realizados de forma local, em um único nó de processamento, mas pode ser facilmente estendido para ambientes com múltiplos nós de processamento.

**#2. HARDWARE: Processador Intel i5-4210U; 1,7 GHz; 2 Núcleos; 4 Threads; 12 GB RAM;
SOFTWARES: Linux CentOS 7; Kernel 3.10; Java 1.8.0.292-b10; Eclipse Java;**

Analise dos Experimentos

DATASET: KAGGLE - URBANSOUND8K

Neste experimento usamos o Dataset UrbanSound8K.

(1) O dataset contém 10 categorias de sons:

air_conditioner, car_horn, children_playing, dog_bark, drilling, engine_idling, gun_shot, jackhammer, siren, street_music

(2) Na fase de Treinamento usamos 30 sons da categoria gun_shot.

(3) Na fase de Teste usamos 10 sons da categoria gun_shot e estabelecemos os Threshold (minimo e maximos).

(4) Obtivemos 90% de acerto na fase de Teste.

(5) O processamento NAO foi executado ate o final.

Analise dos Experimentos

-- TRAIN --

0 - gun_shot - 7060-6-0-0.wav	15 - gun_shot - 7067-6-0-0.wav
1 - gun_shot - 7060-6-1-0.wav	16 - gun_shot - 7068-6-0-0.wav
2 - gun_shot - 7060-6-2-0.wav	17 - gun_shot - 23161-6-0-0.wav
3 - gun_shot - 7061-6-0-0.wav	18 - gun_shot - 23161-6-1-0.wav
4 - gun_shot - 7062-6-0-0.wav	19 - gun_shot - 24631-6-0-0.wav
5 - gun_shot - 7063-6-0-0.wav	20 - gun_shot - 24632-6-0-0.wav
6 - gun_shot - 7064-6-0-0.wav	21 - gun_shot - 24632-6-1-0.wav
7 - gun_shot - 7064-6-1-0.wav	22 - gun_shot - 24652-6-0-0.wav
8 - gun_shot - 7064-6-2-0.wav	23 - gun_shot - 25037-6-0-0.wav
9 - gun_shot - 7064-6-3-0.wav	24 - gun_shot - 25037-6-1-0.wav
10 - gun_shot - 7064-6-4-0.wav	25 - gun_shot - 25038-6-0-0.wav
11 - gun_shot - 7064-6-5-0.wav	26 - gun_shot - 25038-6-1-0.wav
12 - gun_shot - 7065-6-0-0.wav	27 - gun_shot - 25039-6-0-0.wav
13 - gun_shot - 7066-6-0-0.wav	28 - gun_shot - 25039-6-1-0.wav
14 - gun_shot - 7066-6-1-0.wav	29 - gun_shot - 34708-6-0-0.wav

-- TEST --

0 - gun_shot - 34708-6-1-0.wav	8 - gun_shot - 37236-6-0-0.wav
Score: 11.727783203125	Score: 1.496337890625
1 - gun_shot - 34708-6-2-0.wav	9 - gun_shot - 46654-6-0-0.wav
Score: 11.727783203125	Score: 9.52587890625
2 - gun_shot - 34708-6-3-0.wav	
Score: 11.727783203125	
3 - gun_shot - 34708-6-4-0.wav	
Score: 11.727783203125	
4 - gun_shot - 34708-6-5-0.wav	
Score: 11.727783203125	
5 - gun_shot - 35799-6-0-0.wav	
Score: 10.4833984375	
6 - gun_shot - 35800-6-0-0.wav	
Score: 10.408447265625	
7 - gun_shot - 36403-6-0-0.wav	
Score: 9.4599609375	

FAIXA DE 9 A 12
TOTAL 90% ACERTO
NOS TESTES

Analise dos Experimentos

-- CLASSIFY ALL DATASET --

1004 - dog_bark - 344-3-5-0.wav

Score: 11.727783203125

[GUN_SHOT]

*** Next audio in 3 seconds...

FALHA!
IDENTIFICACAO DE LATIDO
COMO TIRO

```
<?xml version="1.0" encoding="ISO-8859-1"?><Request> <RequestId><![CDATA[20220926_160724_244070]]></RequestId> <Command><![CDATA[REQMSG_SENDDATA]]></Command> <Params> <SessionUUID><![CDATA[f84c073363194ef095fb427aff427a8f]]></SessionUUID> <FileName><![CDATA[344-3-5-0.wav]]></FileName> <FileSize><![CDATA[-1]]></FileSize> <FileDataBase64><![CDATA[1]]></FileDataBase64> <Position> <Latitude><![CDATA[22.756206758268867]]></Latitude> <Longitude><![CDATA[44.640764744810966]]></Longitude> </Position></Request>
```

BASEMSG - MessageId: 20220926_160724_244070,MessageTypeId: -1,MessageType: REQMSG_SENDDATA,statusCode: -1,statusMessage: ?

SendData - Request FileName: 344-3-5-0.wav;Latitude: 22.756207;Longitude: 44.640765

```
<?xml version="1.0" encoding="ISO-8859-1"?><Response> <ResponseId><![CDATA[20220926_160724_244089]]></ResponseId> <Response><![CDATA[RESPMSG_SENDDATA]]></Response> <Status> <StatusCode><![CDATA[-1]]></StatusCode> <StatusMessage><![CDATA[?]]></StatusMessage> </Status> <ResponseData> <SessionUUID><![CDATA[f84c073363194ef095fb427aff427a8f]]></SessionUUID> <FileUUID><![CDATA[03912ef124364085991191b55c20e8a8]]></FileUUID> </ResponseData></Response>
```

BASEMSG - MessageId: 20220926_160724_244089,MessageTypeId: -1,MessageType: RESPMSG_SENDDATA,statusCode: -1,statusMessage: ?

SendData - Response FileUUID: 03912ef124364085991191b55c20e8a8

-- CLASSIFY ALL DATASET --

1005 - drilling - 518-4-0-0.wav

Score: 4.0380859375

[?]

SUCESSO!
IDENTIFICACAO DE LATIDO
COMO NAO TIRO

Analise dos Experimentos

-- CLASSIFY ALL DATASET --

1045 - gun_shot - 7060-6-1-0.wav

Score: 10.196044921875

[GUN_SHOT]

*** Next audio in 8 seconds...

```
<?xml version="1.0" encoding="ISO-8859-1"?><Request><RequestId><![CDATA[20220926_161224_544555]]></RequestId> <Command><![CDATA[REQMSG_SENDDATA]]></Command> <Params><SessionUUID><![CDATA[f84c073363194ef095fb427aff427a8f]]></SessionUUID> <FileName><![CDATA[7060-6-1-0.wav]]></FileName> <FileSize><![CDATA[1]]></FileSize> <FileDataBase64><![CDATA[-1]]></FileDataBase64></Params> <Position><Latitude><![CDATA[26.878185216180096]]></Latitude> <Longitude><![CDATA[43.66681944895175]]></Longitude></Position></Request>
```

BASEMSG - MessageId: 20220926_161224_544555,MessageTypeId: -1,MessageType: REQMSG_SENDDATA,statusCode: -1,statusMessage: ?

SendData - Request FileName: 7060-6-1-0.wav;Latitude: 26.878185;Longitude: 43.666819

```
<?xml version="1.0" encoding="ISO-8859-1"?><Response> <ResponseId><![CDATA[20220926_161224_544573]]></ResponseId> <Response><![CDATA[RESPMSG_SENDDATA]]></Response> <Status><StatusCode><![CDATA[-1]]></StatusCode> <StatusMessage><![CDATA[?]]></StatusMessage> </Status> <ResponseData><SessionUUID><![CDATA[f84c073363194ef095fb427aff427a8f]]></SessionUUID> <FileUID><![CDATA[aa5039ea9e8b414780820ed6c8c7481d]]></FileUID> </ResponseData></Response>
```

BASEMSG - MessageId: 20220926_161224_544573,MessageTypeId: -1,MessageType: RESPMSG_SENDDATA,statusCode: -1,statusMessage: ?

SendData - Response FileUID: aa5039ea9e8b414780820ed6c8c7481d

-- CLASSIFY ALL DATASET --

1046 - gun_shot - 7060-6-2-0.wav

Score: 9.878662109375

[GUN_SHOT]

*** Next audio in 9 seconds...

```
<?xml version="1.0" encoding="ISO-8859-1"?><Request><RequestId><![CDATA[20220926_161234_554222]]></RequestId> <Command><![CDATA[REQMSG_SENDDATA]]></Command> <Params><SessionUUID><![CDATA[f84c073363194ef095fb427aff427a8f]]></SessionUUID> <FileName><![CDATA[7060-6-2-0.wav]]></FileName> <FileSize><![CDATA[1]]></FileSize> <FileDataBase64><![CDATA[-1]]></FileDataBase64></Params> <Position><Latitude><![CDATA[25.238684265758824]]></Latitude> <Longitude><![CDATA[47.36355093327342]]></Longitude></Position></Request>
```

BASEMSG - MessageId: 20220926_161234_554222,MessageTypeId: -1,MessageType: REQMSG_SENDDATA,statusCode: -1,statusMessage: ?

SendData - Request FileName: 7060-6-2-0.wav;Latitude: 25.238684;Longitude: 47.363551

```
<?xml version="1.0" encoding="ISO-8859-1"?><Response> <ResponseId><![CDATA[20220926_161234_554233]]></ResponseId> <Response><![CDATA[RESPMSG_SENDDATA]]></Response> <Status><StatusCode><![CDATA[-1]]></StatusCode> <StatusMessage><![CDATA[?]]></StatusMessage> </Status> <ResponseData><SessionUUID><![CDATA[f84c073363194ef095fb427aff427a8f]]></SessionUUID> <FileUID><![CDATA[ff23859208834c4ba9e34bfd14fc0054]]></FileUID> </ResponseData></Response>
```

BASEMSG - MessageId: 20220926_161234_554233,MessageTypeId: -1,MessageType: RESPMSG_SENDDATA,statusCode: -1,statusMessage: ?

SendData - Response FileUID: ff23859208834c4ba9e34bfd14fc0054

SUCESSO!
IDENTIFICACAO DE TIRO
COMO TIRO

Conclusao

#1. Este estudo apresentou o uso da rede neural sem pesos *WiSARD* na deteccao de sons de emergencia.

#2. A rede neural *WiSARD* foi armazenada em um vetor de 4096 valores Inteiros de 32 bits, porem, nos experimentos podemos verificar que pode ser usada vetores menores e com menor precisao, como vetores de 1024 valores de 8 bits (byte).

#3. Nesta implementacao pudemos analisar os detalhes por tras de uma solucao para emergencia, como por exemplo os valores de coordenadas GPS, que podem ser usados para validar uma ocorrencia em funcao da concentracao de dados em uma mesma regioao.

Duvidas

