# Snoring detection based on Arduino Nano 33 BLE Sense Lite


## Lianglei liu


**GitHub**: https://github.com/liangleiliu-lab/casa0018dl4sn

**Edge impulse**: https://studio.edgeimpulse.com/public/381731/live

## 1. Introduction

In this project, the authors trained deep learning models for snoring detection on the Edge Impulse and converted the models into Arduino libraries for deployment on the Arduino Nano 33 BLE Sense Lite. This enables the detection of snoring events. Snoring is a sleep-related breathing disorder that negatively affects both the quality and length of sleep for individuals who snore.

Chronic sleep deprivation and poor sleep quality can lead to memory loss, poor concentration, mood swings and decreased productivity. In addition, snoring is often associated with sleep apnea syndrome, a potentially serious health problem that can increase the risk of heart disease, stroke, and high blood pressure. Despite this, many people who snore are unaware of it. This is not only because snoring often occurs during the deeper stages of sleep, making it difficult to detect, but also because snoring is mistakenly regarded as a harmless "natural phenomenon" in many cultures. As a result, people often fail to recognize the need to seek medical help or take preventive measures.

## 2. Research question

How can a low-cost, portable device based on deep learning and edge computing be developed for detecting and recording snoring events in real time to help people become aware of and manage snoring and the sleep breathing disorders it can cause?

## 3. Application overview

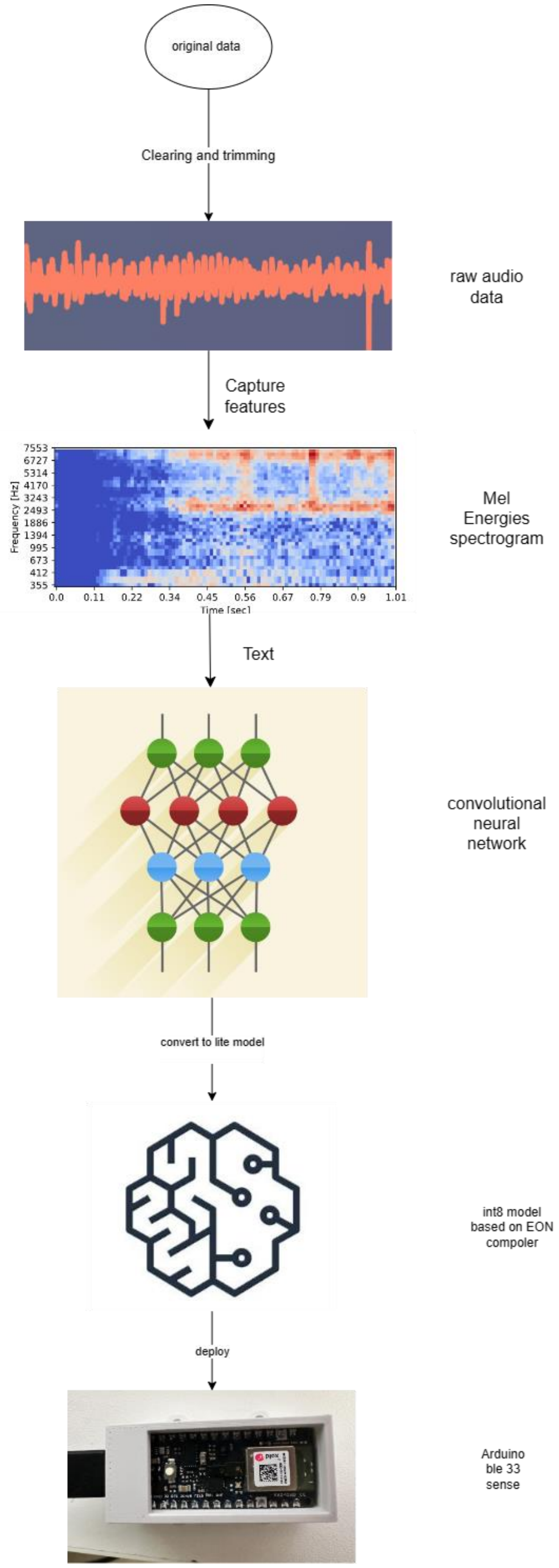



**Figure 1. application diagram**

As can be seen in Figure 1, we have collected many recordings of snoring and noise that may be present in the sleep environment. This was collated into valid samples of one second in duration by cutting and cleaning. After that the authors extracted the Mel filterbank energy features of the audio which are used as inputs to the convolutional neural network. A model that can classify snoring and non-snoring sounds is trained. The model is also optimized for hardware performance based on Arduino Nano 33 BLE sense (cortex-M4F 64 MHz). And deployed to the device.

## 4. Data

**Data source description:**

The dataset employed for this project comprises 1,000 sound samples, divided into two categories: snoring sounds and non-snoring sounds. Each category comprises 500 samples, with a duration of one second. Of the snoring samples, 363 consist of snoring sounds from children, adult males and adult females, with no background sounds. The remaining samples contained non-snoring sounds in the background. The 500 non-snoring samples were divided into ten categories, including babies crying, clocks ticking, doors opening and closing, silence, and the slight sound of a gadget. The background noise included a vibrating motor, toilet flushing, emergency ambulance sirens, thunderstorms, trams, people talking, and TV news in the background. These are considered to be frequently occurring in the scenarios of the application for detecting snoring(khan,2019). I also resampled the audio at 16000Hz to suit the hardware deployment of the product.

**Feature extraction: Mel-Filterbank energy features**

For the analysis of the sound samples, Mel-Filterbank Energy Features (MFE) were extracted. These features are crucial for recognizing various sound patterns and are especially effective in tasks like audio classification, which includes detecting snoring sounds(Agrawal, 2017). MFE provides a representation of the energy distribution across mel-scale frequency bands, capturing essential characteristics relevant to the audio's content.

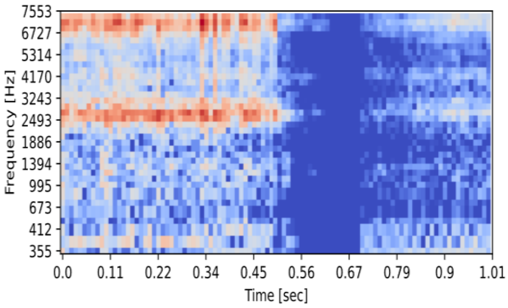


**figure 2. MFE for non-snoring sound.**

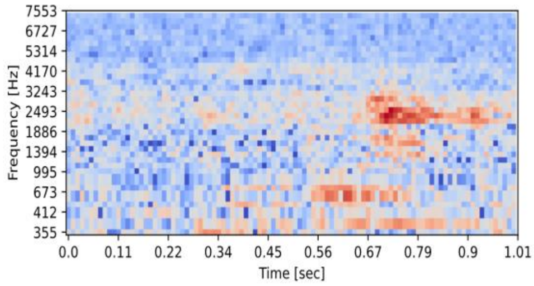


**Figure 3. MFE for snoring sound.**

The Mel-Filterbank Energy (MFE) features of the sound samples are depicted in the figure 2 and figure 3, which illustrate the distinct energy distributions in snoring versus non-snoring sounds.

## 5. Model

**Figure 4. model architecture**

The MFE features extracted from audio signals serve as the input to a neural network specifically designed for snore sound detection. The architecture of this network showing in Figure 4, is tailored to process these audio features effectively through multiple layers:

**Input layer:** receives the vectorized MFE features, initiating the neural processing chain.
**Reshape layer:** transforms the 3960 feature inputs into a 99x40 matrix, aligning the data for temporal analysis.
**1D convolution/Pooling layer:** the first convolution layer with 8 filters and a kernel size of 3 helps extract time dependent local features from sequential frames.
**Dropout layer:** with a dropout rate of 0.25, this layer randomly inactivates certain neurons, enhancing the model's generalization capabilities and reducing overfitting.
**Second 1D convolution/Pooling layer:** a deeper layer with 16 filters, also with a kernel size of 3, captures more complex temporal features.
**Second Dropout layer:** continues to prevent overfitting by inactivating neurons at the same rate of 0.25.
**Flatten layer:** this layer converts the multidimensional data from conventional layers into a one-dimensional vector, preparing it for the final classification step.
**Output layer:** A fully connected layer with two outputs. It classifies the sounds into two categories: snore and non-snore, providing probabilities for each.

The design of this neural network, combined with the precision of MFE feature extraction, offers a robust solution for detecting snore sounds. The use of 1D convolutional layers is particularly advantageous for audio data as it efficiently captures significant features along the time axis. This structure is specifically engineered to handle the temporal and frequency characteristics of sounds like snoring.

## 6. Experiments

| Features input | Numbers of convolution layers(1D) | Dropout | Learning rate | Train cycles | F1 | latency | accuracy | model |
|---|---|---|---|---|---|---|---|---|
| MFE | 2 | 0.25 | 0.005 | 150 | 0.95 | 17ms | 95% | 1 |
| MFE | 2 | 0.2 | 0.005 | 100 | 0.93 | 18ms | 93% | 2 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Spectrum | 2 | 0.5 | 0.005 | 100 | 0.67 | 75ms | 77% | 3 |
| Spectrum | 4 | 0.5 | 0.005 | 100 | 0.69 | 120ms | 75% | 4 |
| Spectrum | 3 | 0.5 | 0.005 | 100 | 0.69 | 65ms | 65% | 5 |
| MFCC | 3 | 0.5 | 0.005 | 100 | 0.67 | 57ms | 67% | 6 |
| MFCC | 2 | 0.5 | 0.005 | 100 | 0.65 | 43ms | 57% | 7 |

**Table1. experiments results**

In my research on snore sound detection, I explored the performance of different feature extraction methods and neural network configuration. The goal was to optimize the model for high accuracy, low latency, and robustness, as represented by the F1 score. The table 1 shows the findings.

**MFE-based models:**

**Model1:** Using MFE features with two convolution layers and a dropout rate of 0.25, we achieved an F1 score of 0.95 and an impressive accuracy of 95% with a latency of only 17 ms after 150 train cycles. This configuration provided the best overall performance.

Model 2: Reducing the dropout rate slightly to 0.2 and train cycles to 100, the F1 score slightly decreased to 0.93, with a marginal increase in latency (18 ms), and an accuracy of 93%.

Spectrum-based models:

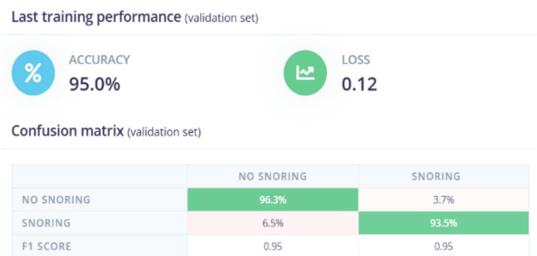
**Model 3-5:** Utilizing Spectrum features, author tested models with varying numbers of convolution layers. All models maintained the same dropout rate of 0.5 and learning rate. The model with two convolution layers reached an F1 score of 0.67, latency of 75 ms, and 77% accuracy. Increasing convolution layers to four slightly improved the F1 score to 0.69 but resulted in higher latency (120 ms) and reduced accuracy (75%). A configuration with three layers had a similar F1 score but improved latency (65 ms), though at a lower accuracy of 65%.

**MFCC-based models:**

**Model 6-7:** These models employed MFCC features. A configuration with three convolution layers resulted in an F1 score of 0.67, latency of 57 ms, and 67% accuracy. Reducing the convolution layers to two produced a lower F1 score of 0.65, but the latency improved significantly to 43 ms; however, the accuracy dropped to 57%.

The MFE-based models demonstrated superior performance in both accuracy and latency, making them highly suitable for real-time snore detection applications. In contrast, models using Spectrum and MFCC features showed varied results, with generally lower accuracy and higher latencies, indicating that while they may be useful in less time-sensitive applications.

## 7. Results and observations



Last training performance (validation set)

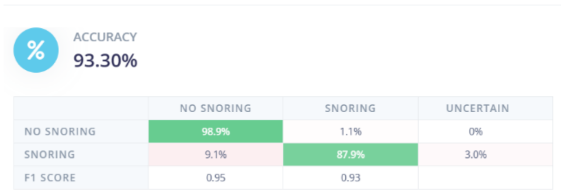
ACCURACY 95.0%

LOSS 0.12

Confusion matrix (validation set)

| | NO SNORING | SNORING |
|---|---|---|
| NO SNORING | 96.3% | 3.7% |
| SNORING | 6.5% | 93.5% |
| F1 SCORE | 0.95 | 0.95 |

Figure.5 validation set test result.



ACCURACY 93.30%

| | NO SNORING | SNORING | UNCERTAIN |
|---|---|---|---|
| NO SNORING | 98.9% | 1.1% | 0% |
| SNORING | 9.1% | 87.9% | 3.0% |
| F1 SCORE | 0.95 | 0.93 | |

Figure.6 general test result.

In the validation set test results in Figure 5 and the generalised test results in Figure 6, the model consistently produces good predictions, but there are still discrepancies worth discussing. Below is a comparison of these two datasets:

**Accuracy**

- **Validation Set**: 95.0%
- **General Test**: 93.30%

The model exhibits slightly higher accuracy on the validation set, likely due to the more ideal data environment without additional noise interference. Despite the added noise in the general test, the model maintains a commendable accuracy of 93.30%, demonstrating good resistance to noise.

**Confusion Matrix**

**No Snoring**

- **Validation Set**: Correctly classified as no snoring at 96.3%, misclassified as snoring at 3.7%.

- **General Test**: Correctly classified as no snoring at 98.9%, misclassified as snoring at 1.1%.

For no snoring classifications, the general test performs better, with a lower misclassification rate. This suggests that in noisy environments, the model is more adept at accurately identifying instances without snoring.

**Snoring**

- **Validation Set**: Correctly classified as snoring at 93.5%, misclassified as no snoring at 6.5%.
- **General Test**: Correctly classified as snoring at 87.9%, misclassified as no snoring at 9.1%, with 3.0% uncertain.

In terms of snoring classifications, the validation set outperforms the general test. The presence of noise might mask some characteristics of snoring sounds, leading to a lower recognition accuracy.

**F1 Score**

- **Validation Set**: Both no snoring and snoring categories scored 0.95.
- **General Test**: No snoring scored 0.95, snoring scored 0.93.

The F1 scores are relatively high in both tests, indicating that the model achieves a good balance in precision and recall. However, the F1 score for snoring in the general test is slightly lower, again highlighting potential challenges in complex environments.

## Bibliography

1. T. H. Khan, "A deep learning model for snoring detection and vibration notification using a smart wearable gadget," Electronics, vol. 8, no. 9, article. 987, ISSN 2079-9292, 2019.
2. Tak, R.N., Agrawal, D.M. and Patil, H.A., 2017, November. Novel phase encoded mel filterbank energies for environmental sound classification. In *International Conference on Pattern Recognition and Machine Intelligence* (pp. 317-325). Cham: Springer International Publishing.