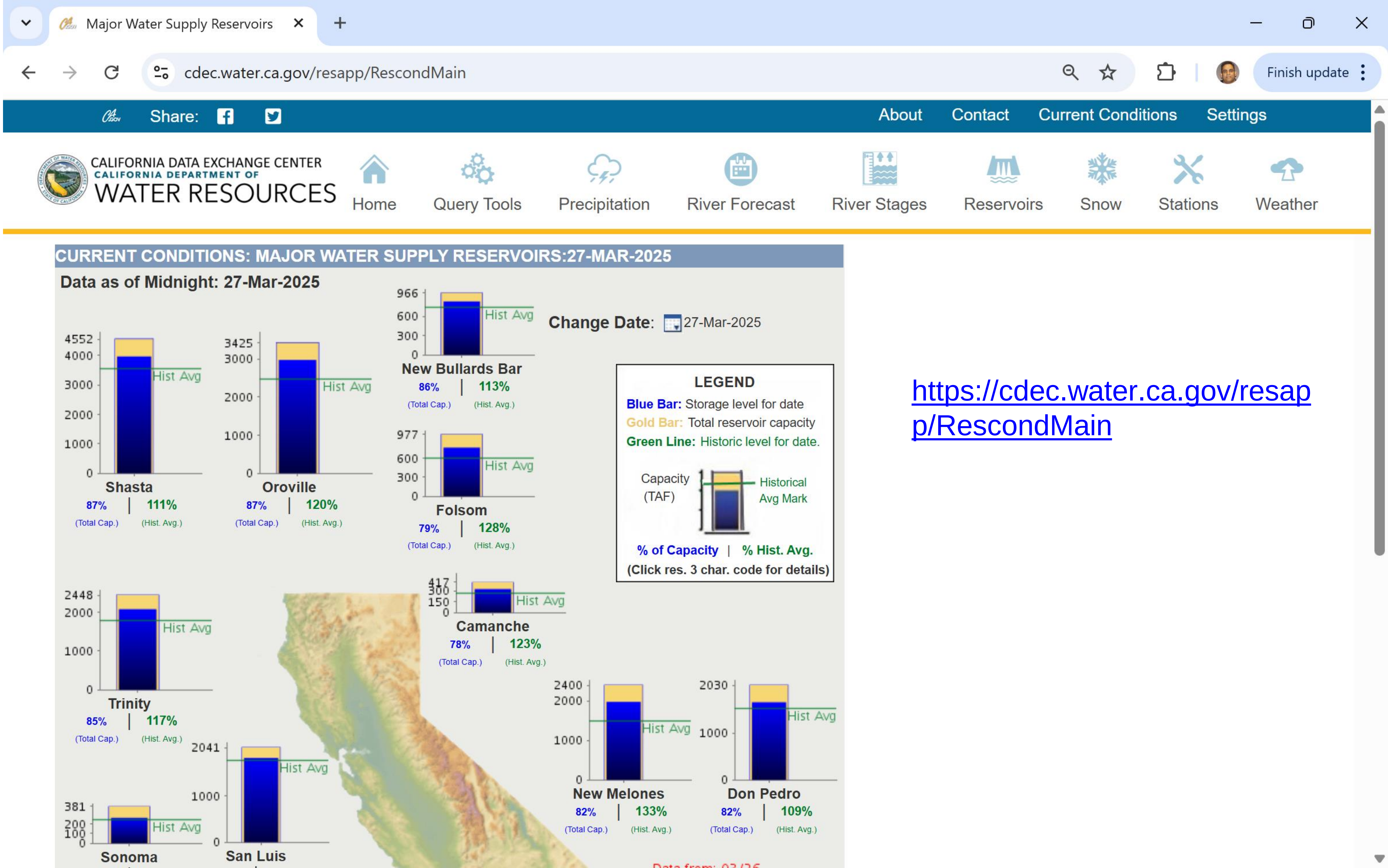
CMPE 273 Hackathon – California Reservoirs Data & Car Sounds Dashboard



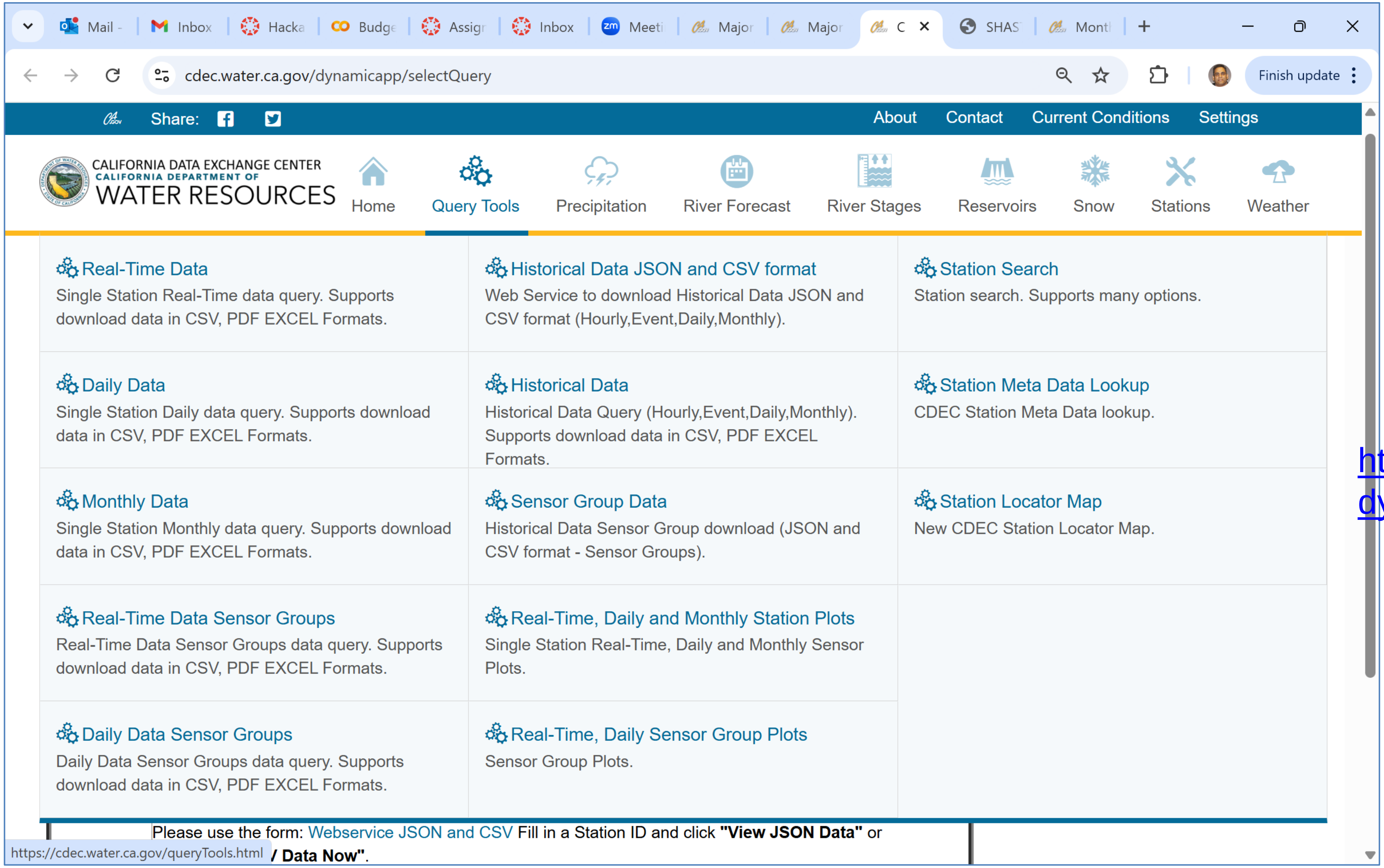
https://cdec.water.ca.gov/resapp/RescondMain

Dataset

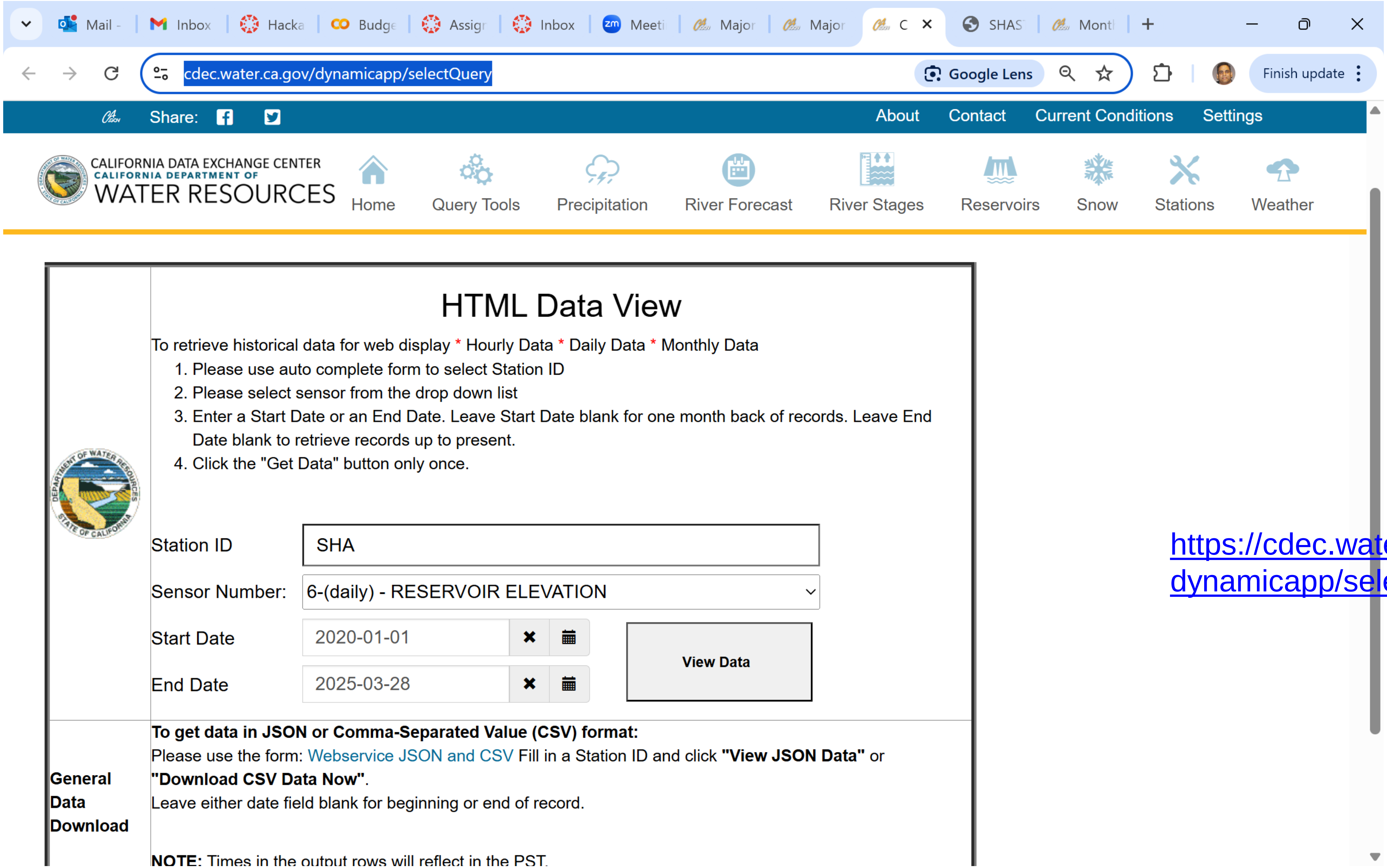
California's major water supply reservoirs are identified by unique codes assigned by the California Data Exchange Center (CDEC). These codes are used for monitoring and reporting purposes. Here are some of the largest reservoirs along with their corresponding CDEC IDs:

Reservoir Name	CDEC ID
Shasta Lake	SHA
Lake Oroville	ORO
Trinity Lake	CLE
New Melones Lake	NML
San Luis Reservoir	SNL
Don Pedro Reservoir	DNP
Lake Berryessa	BER
Folsom Lake	FOL
New Bullards Bar Reservoir	BUL
Pine Flat Lake	PNF

Historical Dataset



https://cdec.water.ca.gov/dynamicapp/selectQuery

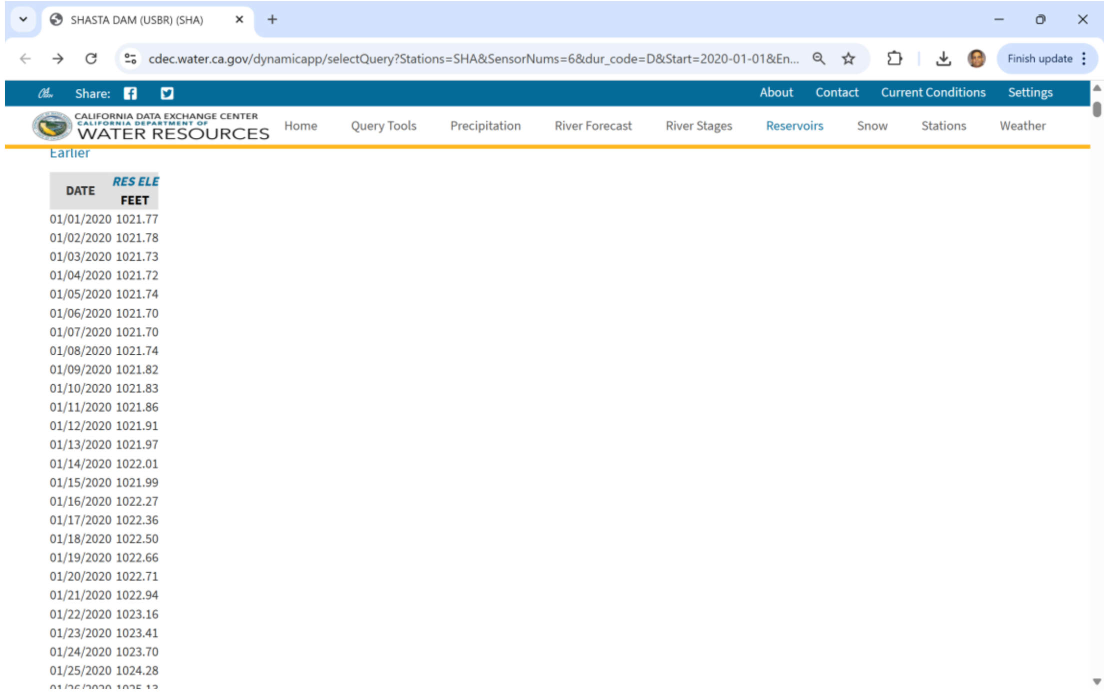


https://cdec.water.ca.gov/dynamicapp/selectQuery

Historical Dataset

Elevation: 1067' · SACRAMENTO RIVER basin ·
Operator: US Bureau of Reclamation
River Stage Definitions:

Query executed Friday at 10:17:03

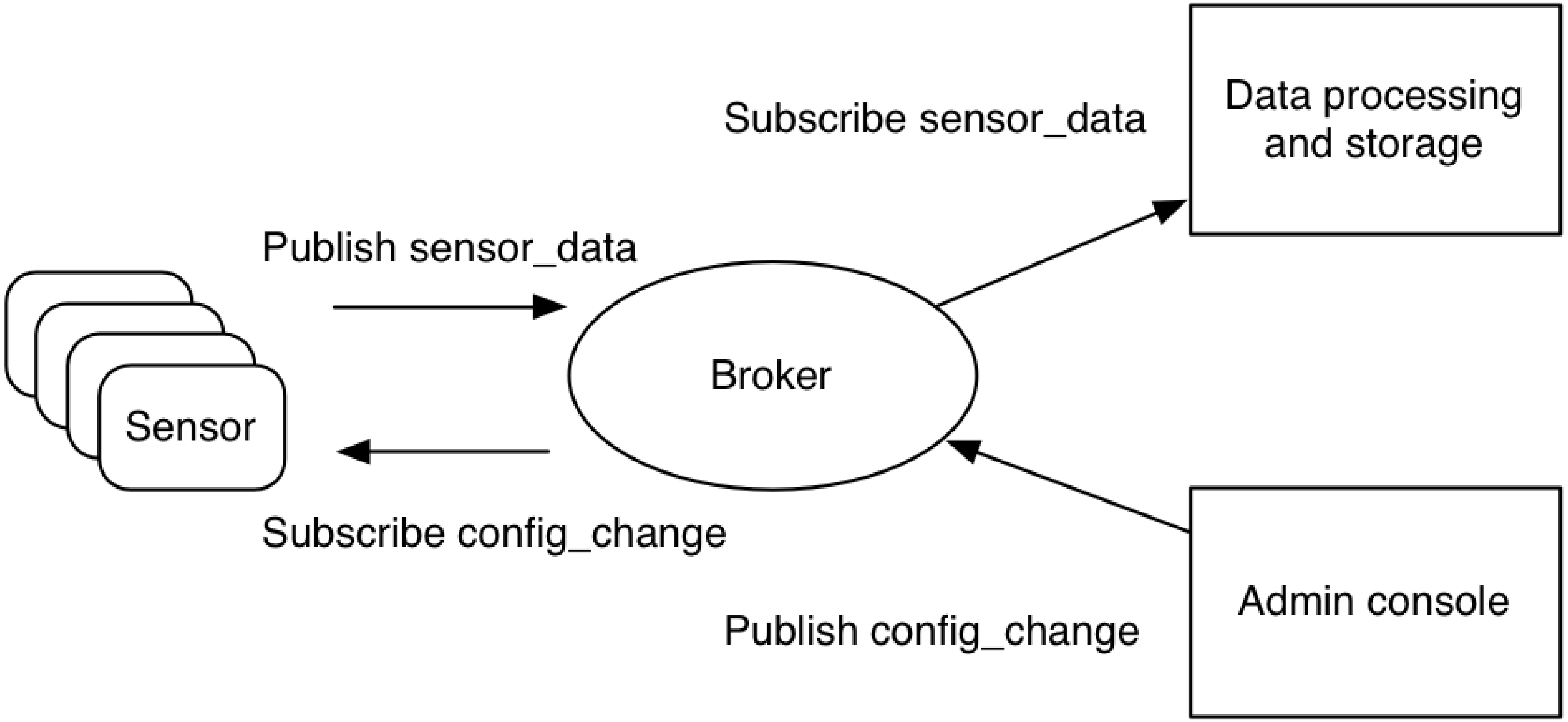


https://cdec.water.ca.gov/dynamicapp/selectQuery?Stations=SHA&SensorNums=6&dur_code=D&Start=2020-01-01&End=2025-03-28

Historical Dataset

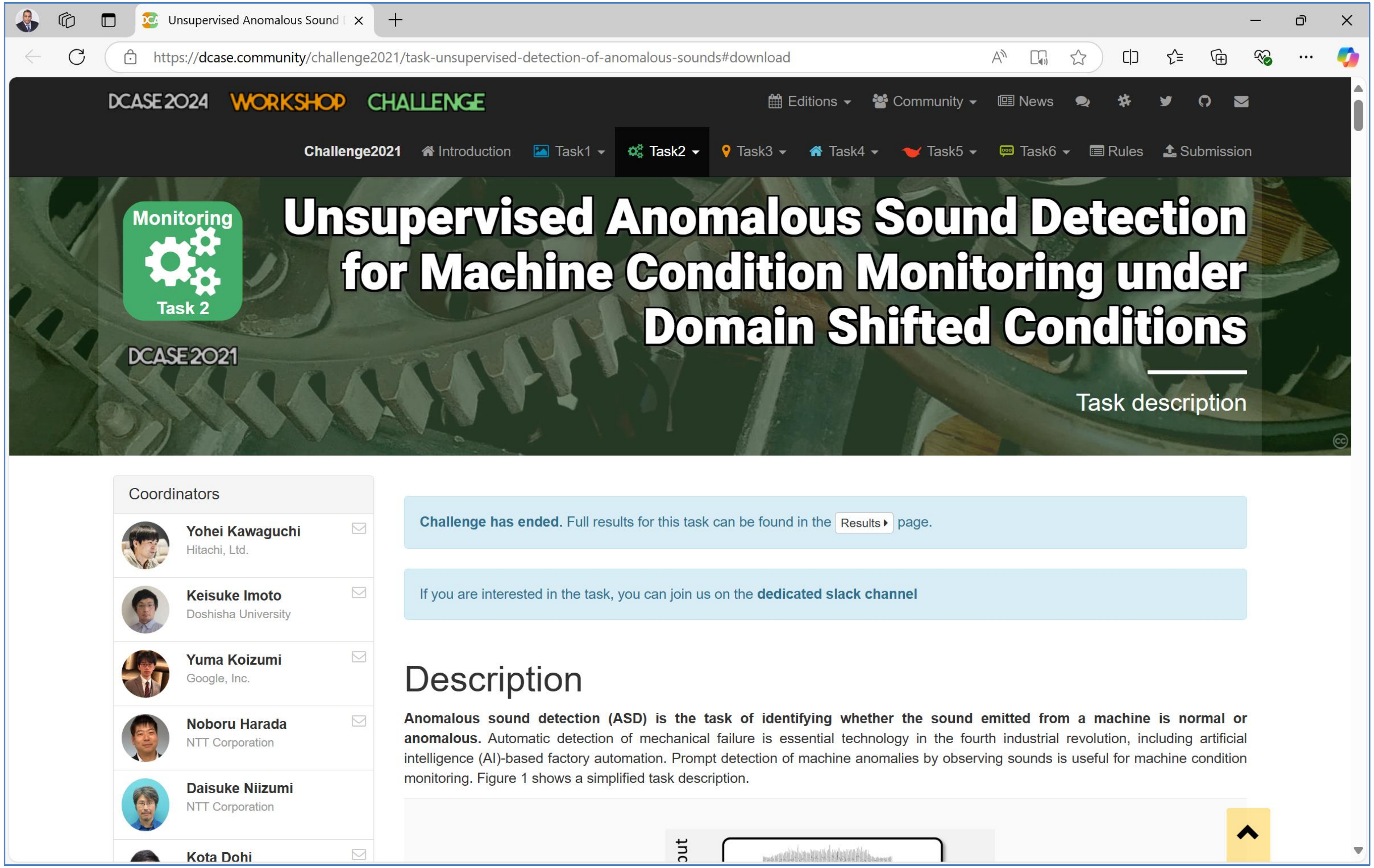
Create MQTT Data Architecture to inject data into Topics. Process data and present it in output.

Make sure MQTT Topic is separated by station.



Car Sounds ML Model

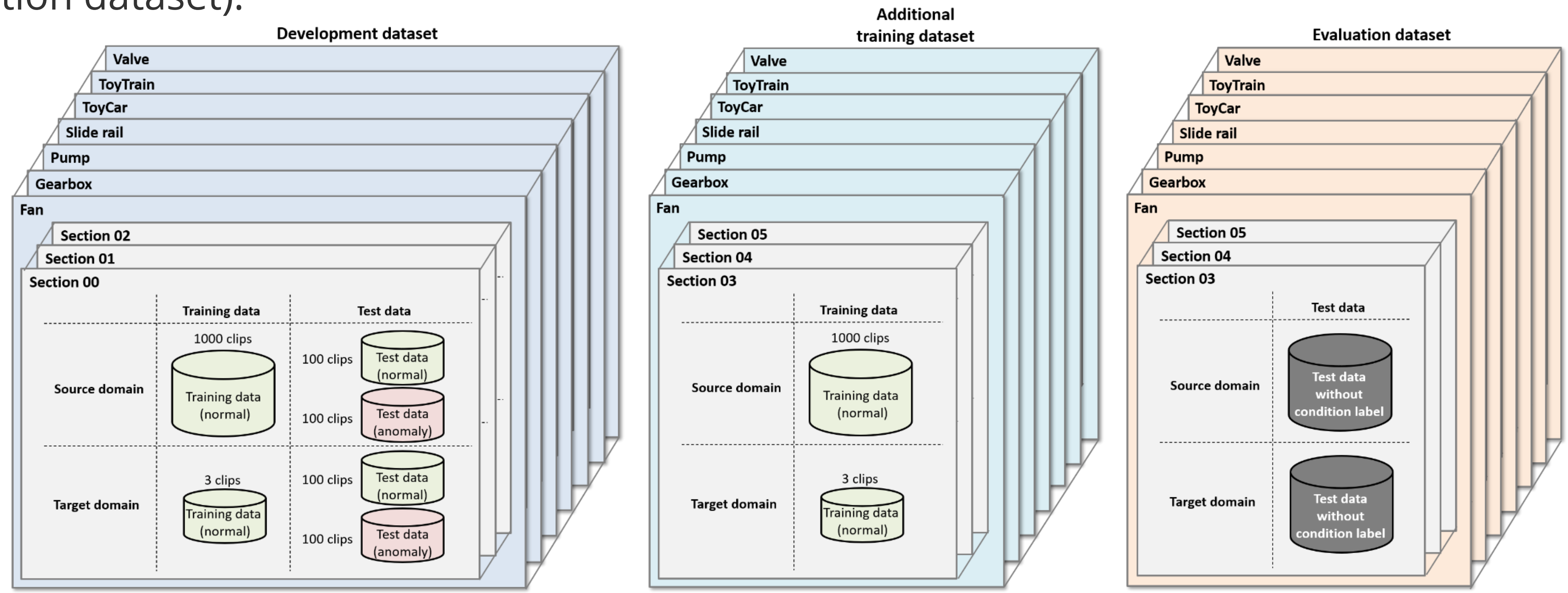
Dataset



Source: [Unsupervised Anomalous Sound Detection for Machine Condition Monitoring under Domain Shifted Conditions - DCASE]

DCASE 2021 Challenge Task 2 Development Dataset

Figure shows an overview of the datasets: the development, additional training, and evaluation datasets. Every dataset consists of the seven types of machines, and each type of machine consists of three "sections" (Section 00, 01, and 02 for the development dataset and Section 03, 04, and 05 for the additional training dataset and the evaluation dataset).



Download

- [Development dataset (7.5 GB)]
- [Additional training dataset (5.5 GB)]
- [Evaluation dataset (2.2 GB)]

Source: Unsupervised Anomalous Sound Detection for Machine Condition Monitoring under Domain Shifted Conditions - DCASE

Dataset

Sounds carry a wealth of information about our everyday surroundings and the physical events that occur within them. We have the ability to perceive the auditory landscape we inhabit—whether it's a bustling street, an office, or another setting—and to identify individual sound sources, such as a car passing by or footsteps approaching. The development of signal processing methods to automatically extract this information holds immense potential across various applications. These include searching multimedia based on audio content, creating context-aware mobile devices, enhancing the capabilities of robots and cars, and implementing intelligent monitoring systems that recognize activities in their environments through acoustic information.

However, substantial research is still required to reliably recognize sound scenes and individual sound sources in real-world soundscapes. These environments often feature multiple overlapping sounds that are frequently distorted by the surroundings [1, 2].

DCASE2021 Challenge: https://dcase.community/challenge2021/index

Source: https://odin.tradoc.army.mil/WEG/Asset/755c0e281c53067c01a14af195f76eaf

Feature Extraction

Extracting audio features and representations, such as Mel-Frequency Cepstral Coefficients (MFCCs), spectrograms, and chromagrams, forms the foundation for advanced audio analysis. This process involves training, parameter tuning, and evaluating classifiers for audio segments, enabling the classification of unknown sounds. Additionally, the detection of audio events and the exclusion of silence periods from lengthy recordings are crucial for accurate analysis. Advanced techniques such as supervised segmentation, which combines segmentation and classification, and unsupervised segmentation, exemplified by speaker diarization, facilitate the extraction of audio thumbnails. Training and utilizing audio regression models, particularly for applications like emotion recognition, further enhance the analytical capabilities. Moreover, applying dimensionality reduction techniques allows for the visualization of audio data and the exploration of content similarities, offering deeper insights into the auditory landscape [3, 4].

Source: https://odin.tradoc.army.mil/WEG/Asset/755c0e281c53067c01a14af195f76eaf

Feature Extraction

Several frames to sounds features:

- Time domain features include:
- Zero Crossing Rate (ZCR) and
- Short Time Energy (STE).

Several frames to sounds features:

Spectral Features Include:

- Linear Predictive Coding (LPC) coefficients,
- Relative Spectral Predictive Linear Coding (RASTA PLP),
- Pitch,
- Sone,
- Spectral Flux (SF) and
- coefficients from basic time to frequency transforms (FFT, DFT, DWT, CWT and Constant Q-Transform)

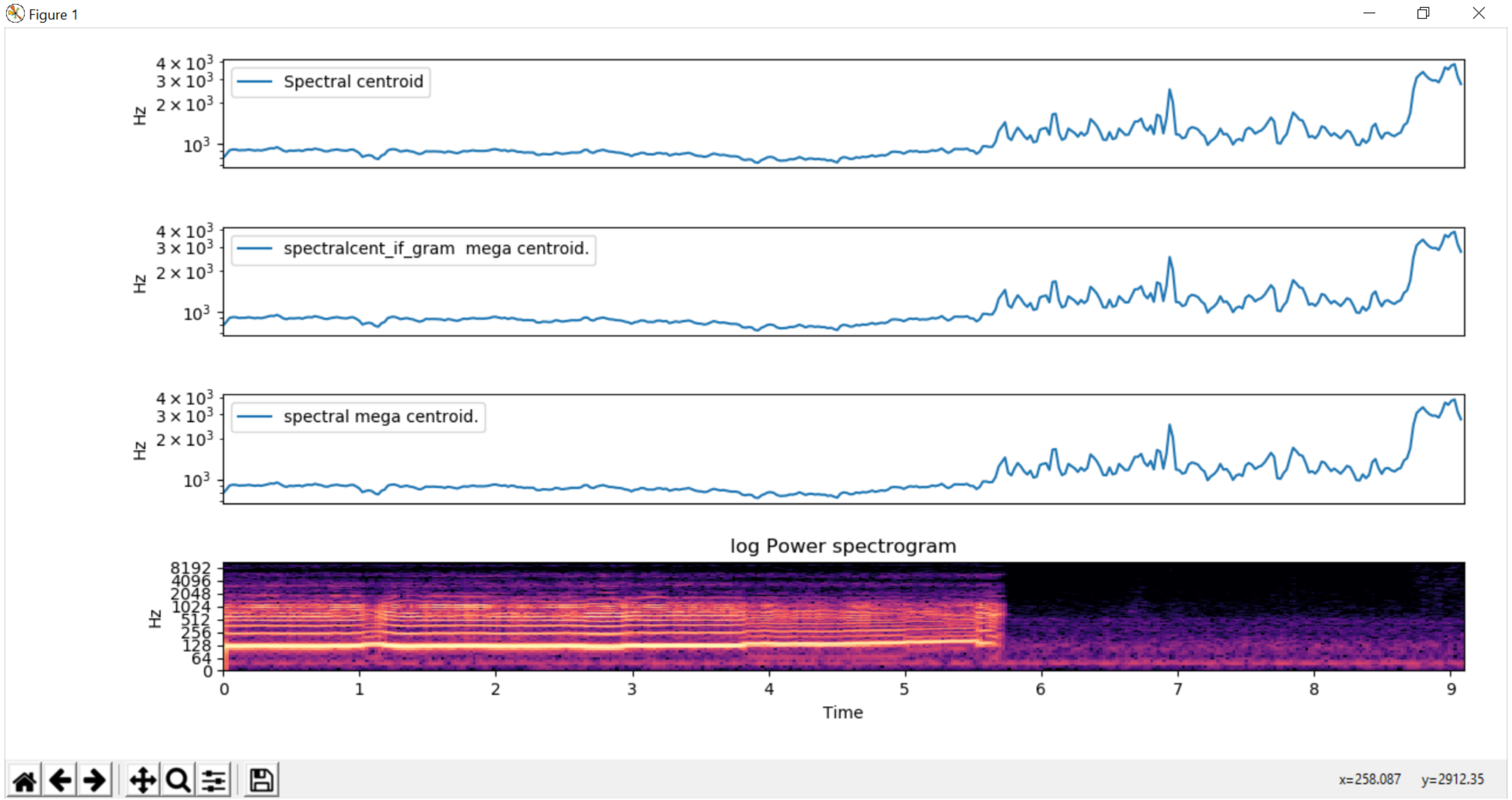
Source: https://odin.tradoc.army.mil/WEG/Asset/755c0e281c53067c01a14af195f76eaf

Feature Extraction

Several frames to sounds features:

Cepstral domain features are

- Mel Frequency Cepstral Coefficient (MFCC) and
- Bark Frequency Cepstral Coefficient (BFCC).

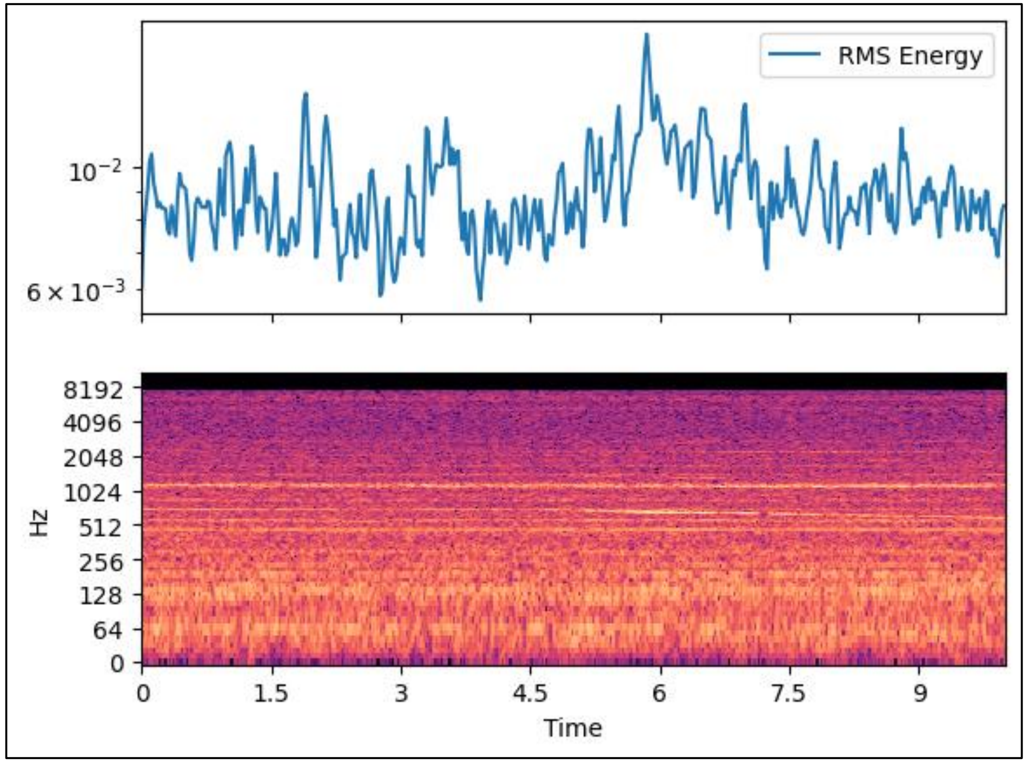


Source: https://odin.tradoc.army.mil/WEG/Asset/755c0e281c53067c01a14af195f76eaf

RMS Energy

The plot shows the Root Mean Square (RMS) energy of an audio signal (please see figure 1) over time. RMS energy is a measure of the power or loudness of the signal at each time frame and is calculated by taking the square root of the mean of the squares of the signal values. The spectrogram (bottom) is a visual representation of the spectrum of frequencies in the audio signal as it varies with time. It is created by applying a Short-Time Fourier Transform (STFT) to the audio signal.

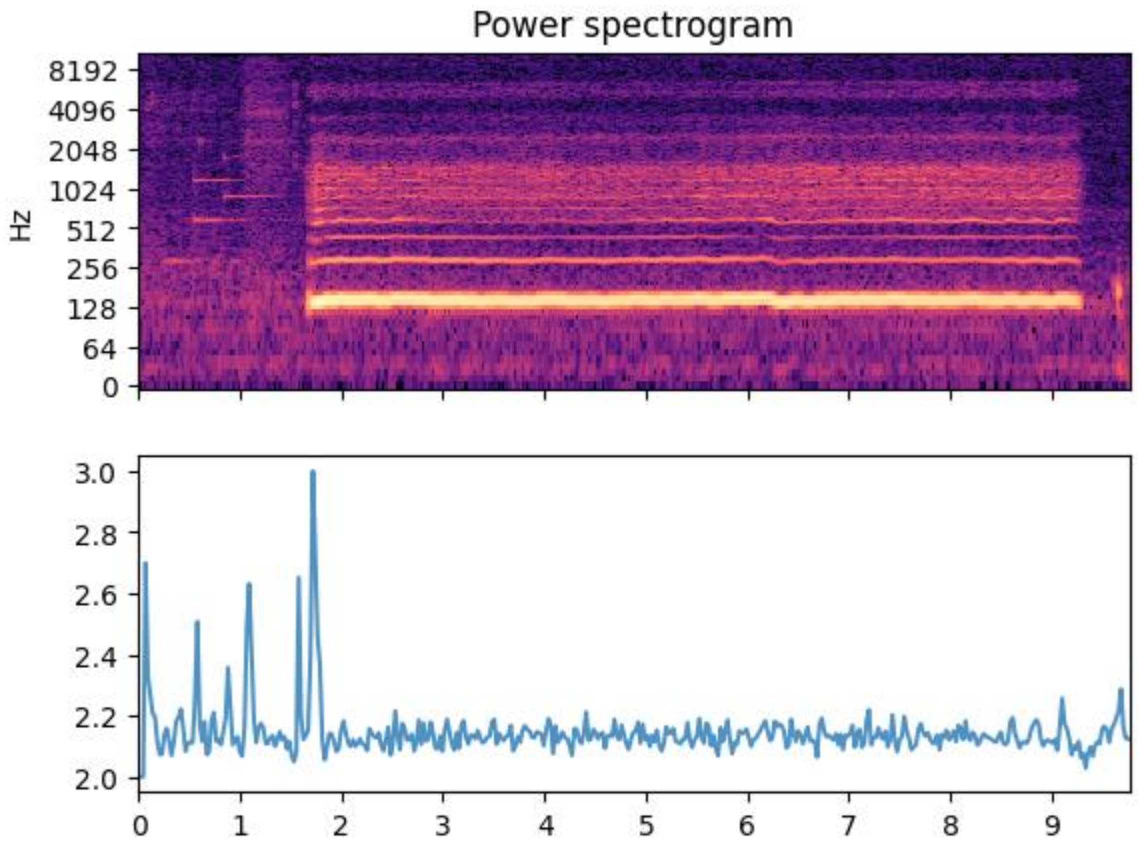
In summary, these plots provide a comprehensive view of Fan audio signal's characteristics, with the RMS energy plot highlighting changes in loudness over time and the spectrogram revealing the detailed frequency content.



Source: https://odin.tradoc.army.mil/WEG/Asset/755c0e281c53067c01a14af195f76eaf

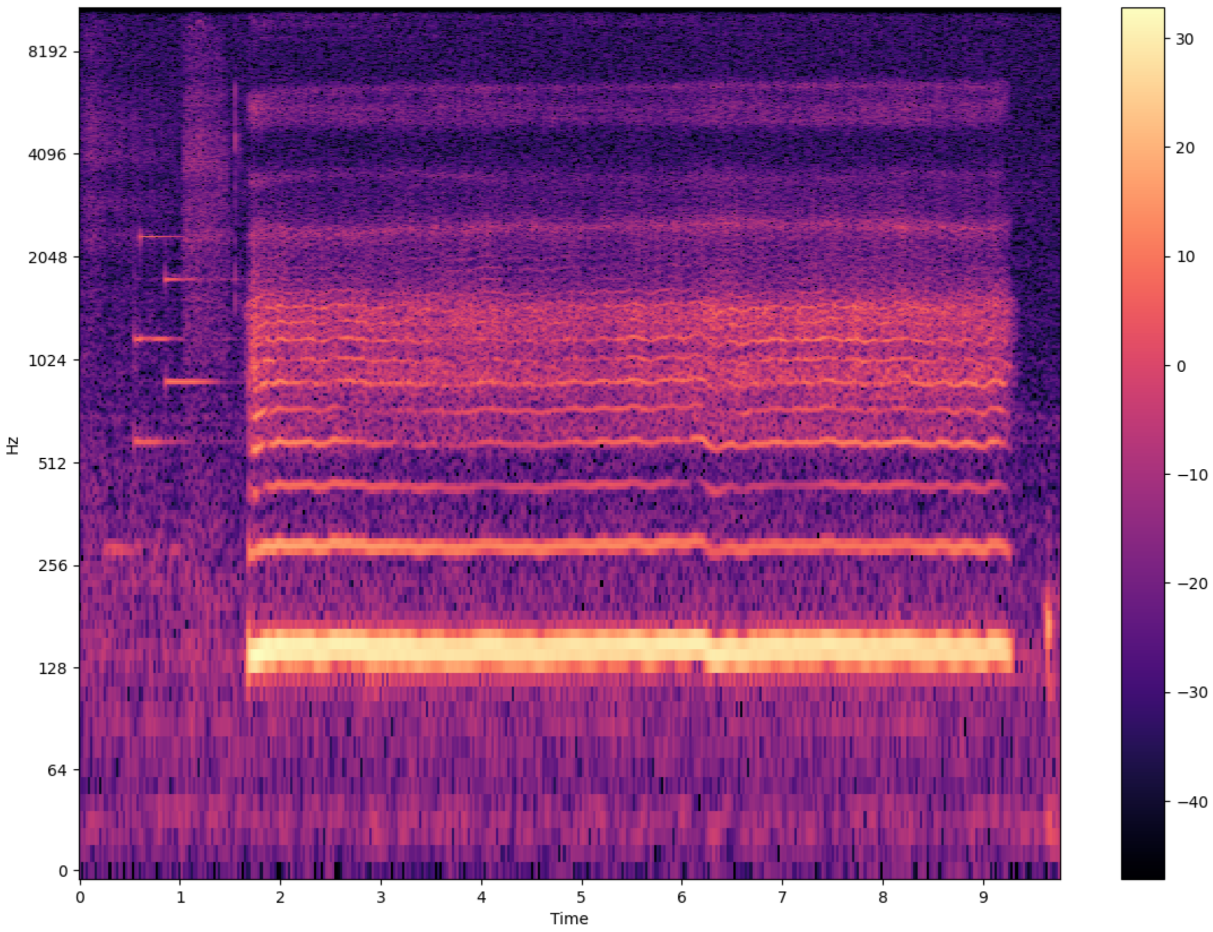
Power Spectrogram

```
D = np.abs(librosa.stft(x))
times = librosa.times_like(D)
fig, ax = plt.subplots(nrows=2, sharex=True)
librosa.display.specshow(librosa.amplitude_to_db(D,
ref=np.max),
                         y_axis='log', x_axis='time', ax=ax[0])
ax[0].set(title='Power spectrogram')
ax[0].label_outer()
onset_env = librosa.onset.onset_strength(y=x, sr=sr)
ax[1].plot(times, 2 + onset_env / onset_env.max(),
alpha=0.8,
         label='Mean (mel)')
```



Power Spectrogram

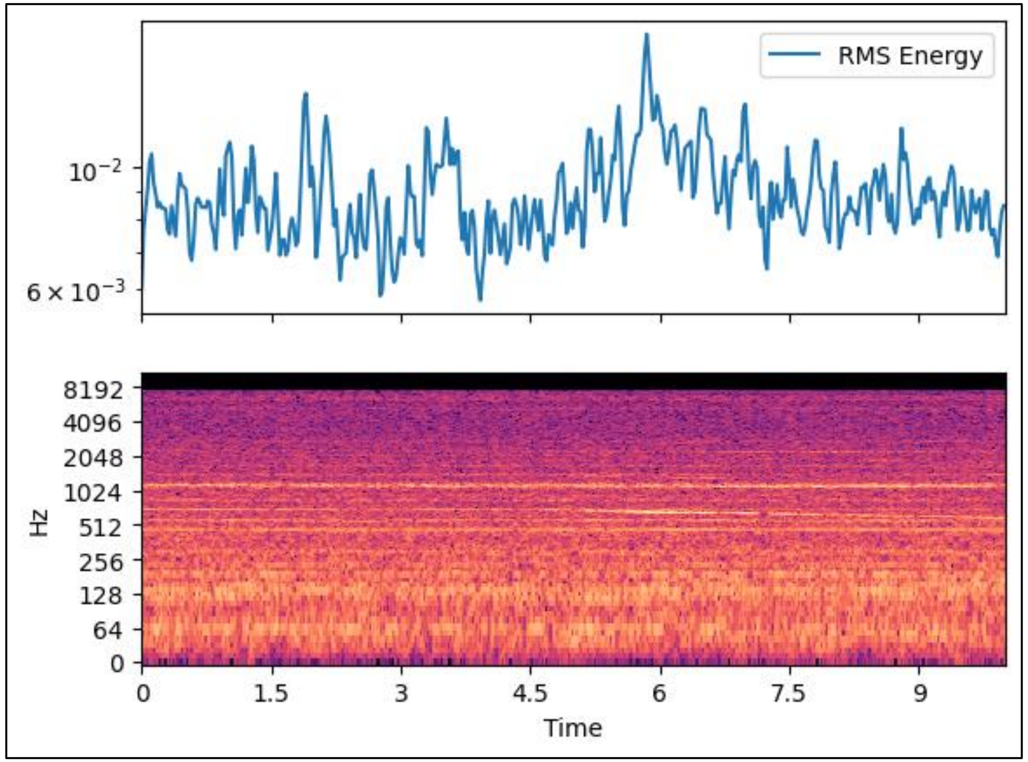
```
import matplotlib.pyplot as plt
X = librosa.stft(x)
Xdb = librosa.amplitude_to_db(abs(X))
plt.figure(figsize=(14, 10))

librosa.display.specshow(Xdb, sr=sr, x_axis='time', y_axis='log')
plt.colorbar()
```



RMS Energy

MFCC: Mel-Frequency Cepstral Coefficients (MFCCs) is derived from [3, 4] *Signal* → *Pre-emphasis* → *Hamming Window* → *Fast Fourier Transform* → *Log* → *cosine* → *Mel-frequency Cepstral coefficients* →

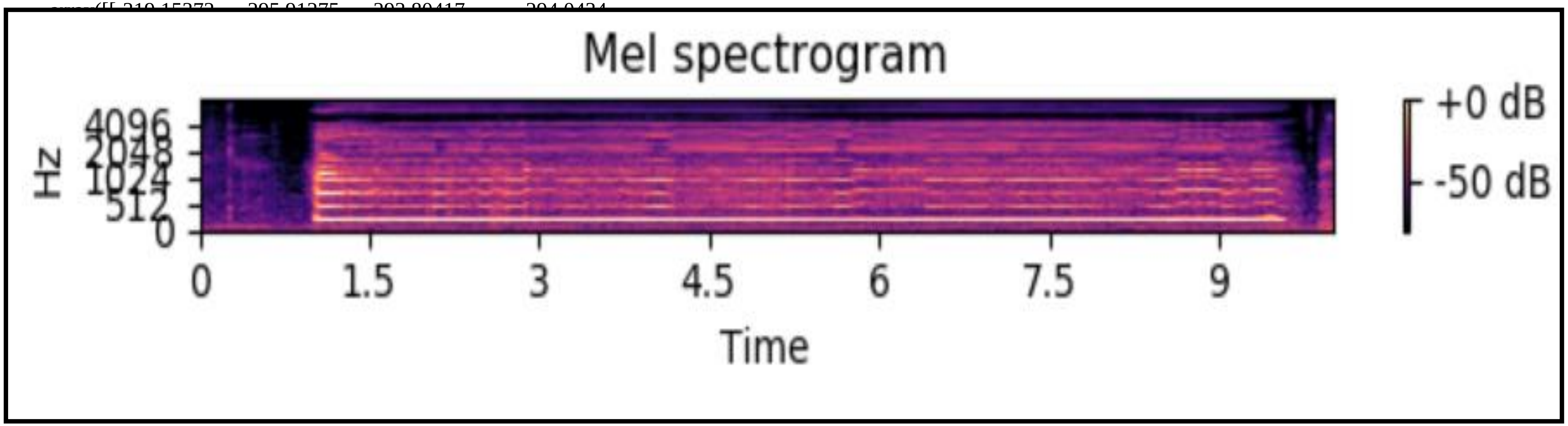
```
array([[-319.15372  , -295.91275  , -293.80417  , ..., -294.0434   ,
        -293.89774  , -291.26505  ],
       [ 134.34085  ,  138.12479  ,  135.21011  , ...,  129.9654   ,
         134.12906  ,  128.42188  ],
       [ -40.449566 ,  -46.675735 ,  -44.36908  , ...,  -43.19303  ,
         -42.587517 ,  -34.726013 ],
       ...,
       [  -7.541944 ,   -9.2912655,   -8.514088 , ...,  -17.529697 ,
         -19.356907 ,   -7.161364 ],
       [  11.281778 ,   12.187988 ,   10.807693 , ...,    2.3839679,
           1.5097455,   -0.36110342],
       [   1.4317584,   -3.484956 ,   -2.6210365, ...,   -9.327881 ,
          -9.596742 ,   -8.223911 ]], dtype=float32)
```



Source: https://odin.tradoc.army.mil/WEG/Asset/755c0e281c53067c01a14af195f76eaf

Mel Spectrogram

mel = librosa.feature.melspectrogram(y=sig, sr=sample_rate)



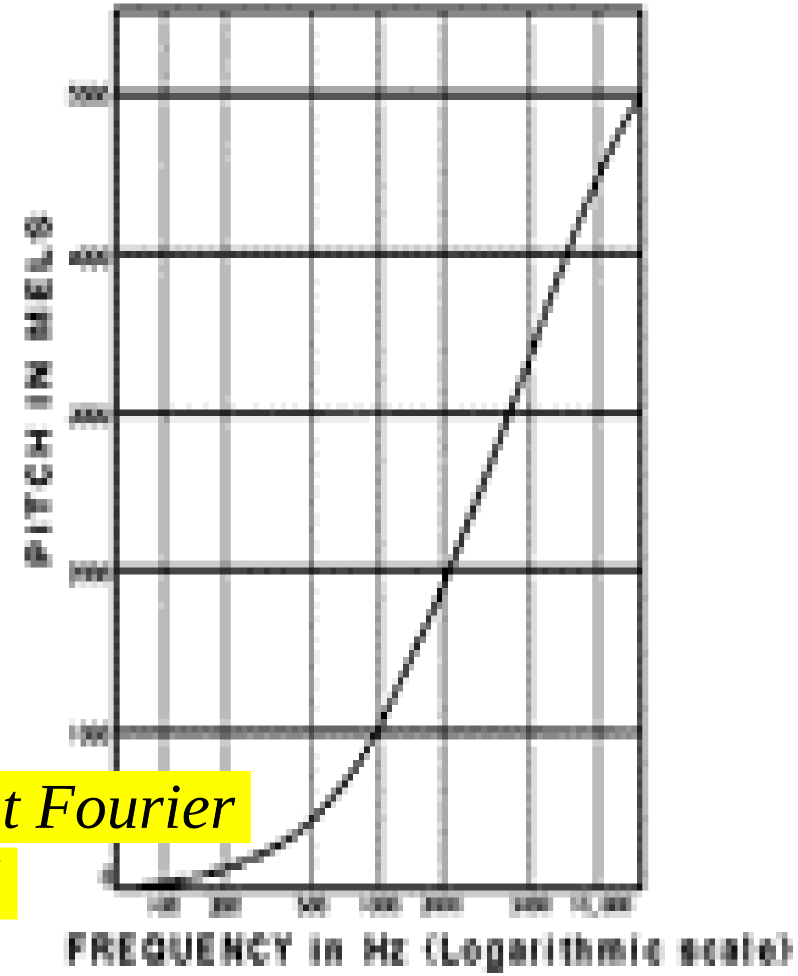

```
1.5097455 ,   -0.36110342],
[   1.4317584 ,   -3.484956  ,   -2.6210365 , ...,   -9.327881  ,
   -9.596742  ,   -8.223911  ]], dtype=float32)
```

Source: https://odin.tradoc.army.mil/WEG/Asset/755c0e281c53067c01a14af195f76eaf

Mel Spectrogram

MFCC
The mel-frequency cepstrum (MFC) is the representation of short-term power spectrum of a sound that is derived from a linear cosine transform of a log power spectrum on a nonlinear of a mel scale of frequency.

Please note mel scale is a scale of pitches with reference point set to 1000 Hz tone, 40 dB above listener's threshold, with a pitch of 1000 mels



Signal → Pre-emphasis → Hamming Window → Fast Fourier Transform → Log → cosine → Mel-frequency Cepstral coefficients → MFCC

Source: https://odin.tradoc.army.mil/WEG/Asset/755c0e281c53067c01a14af195f76eaf

ML Model: Support Vector Machine (SVM)

Support Vector Machine (SVM) is a powerful supervised learning model commonly used for classification and regression tasks. One of the key features of SVM is its ability to handle non-linear data through the use of kernel functions. The Radial Basis Function (RBF) kernel is one of the most popular and effective kernels used in SVM.

- SVM works by finding the hyperplane that best separates the data into different classes. The data points that are closest to this hyperplane are called support vectors. These points are crucial because they define the position and orientation of the hyperplane.
- The Radial Basis Function (RBF) kernel, also known as the Gaussian kernel, is defined as: [

    K(x, x') = \exp\left(-\gamma |x - x'|^2\right)

    ]

  where (\gamma) is a parameter that defines the influence of a single training example. The RBF kernel maps the input features into a higher-dimensional space, making it easier to find a hyperplane that separates the classes.

Source: https://odin.tradoc.army.mil/WEG/Asset/755c0e281c53067c01a14af195f76eaf

ML Model: Support Vector Machine (SVM)

Support Vector Machine (SVM) is a powerful supervised learning model commonly used for classification and regression tasks. One of the key features of SVM is its ability to handle non-linear data through the use of kernel functions. The Radial Basis Function (RBF) kernel is one of the most popular and effective kernels used in SVM.

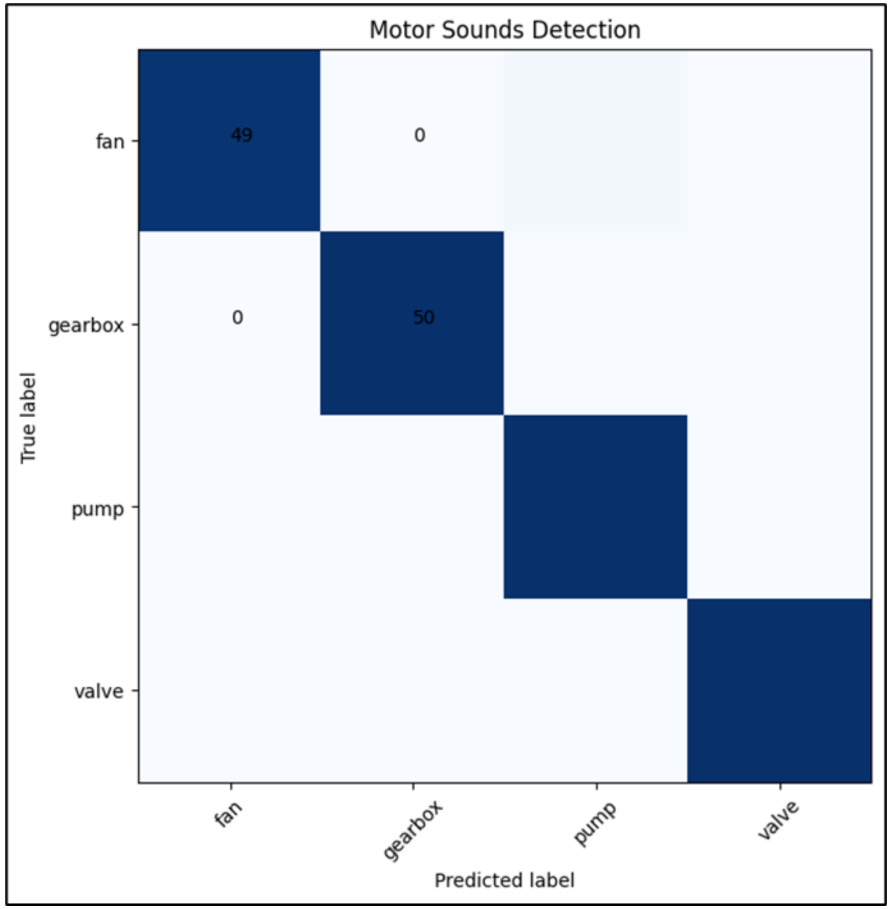
```
# extract short-term features using a 50msec non-overlapping windows
mt, st = 0.05, 0.05

dirs = ["fan", "gearbox", "pump", "valve"]

extract_features_and_train(dirs, mt, mt, st, st, "svm_rbf",
                           "motorsoundsmodel")
```

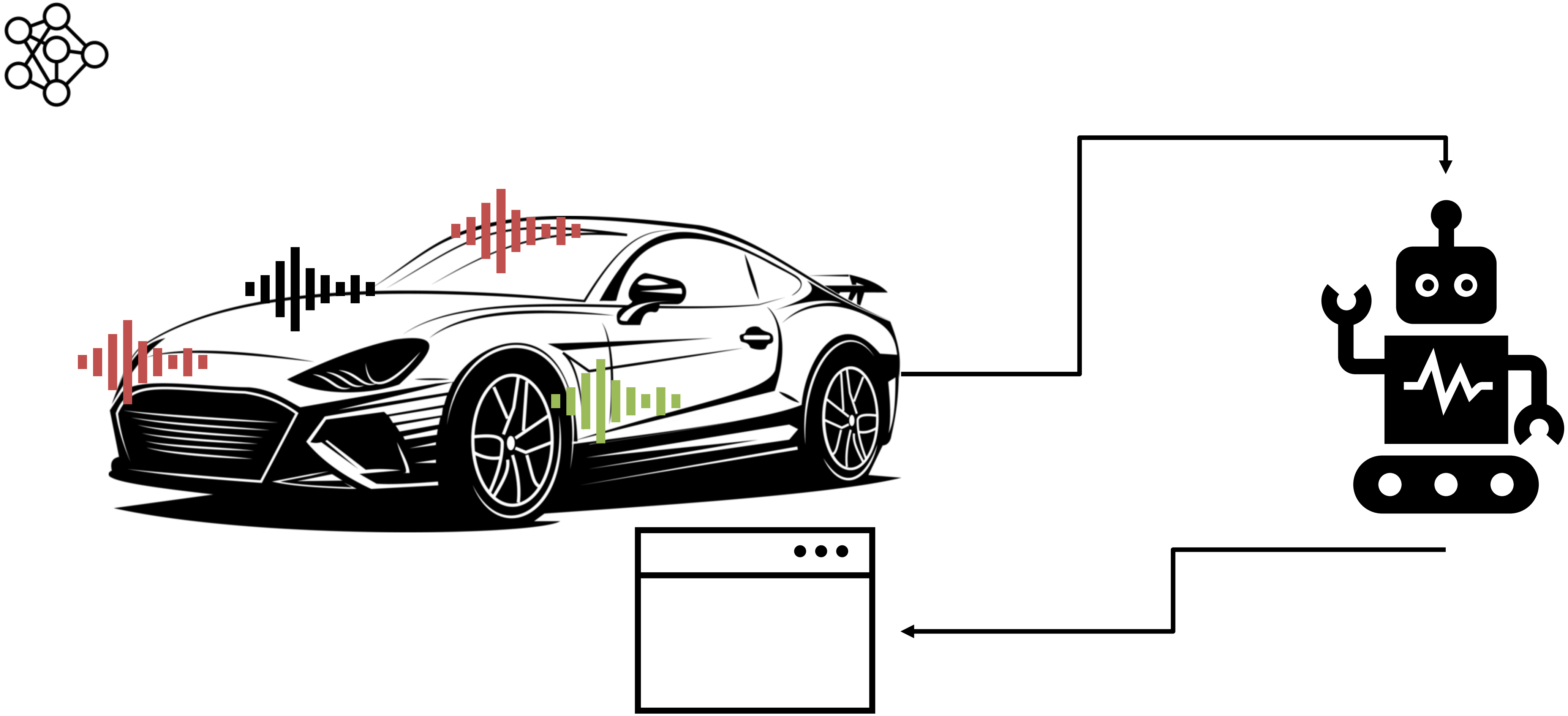
	fan			gearbox			
	pump			valve		OVERALL	
	C	PRE	REC	f1	PRE	REC	f1
	PRE	REC	f1	PRE	REC	f1	
	ACC	f1					
	0.001	15.3	15.7	15.5	15.5	16.7	16.1
	16.8	18.6	17.6	20.6	16.5	18.3	16.9
	16.9						
	0.010	15.1	16.9	16.0	16.5	17.0	16.8
	16.9	15.1	16.0	20.7	19.9	20.3	17.2
	17.2						
	0.500	25.4	17.7	20.8	31.0	29.7	30.3
	31.3	47.5	37.7	90.1	77.3	83.2	42.7
	43.0						
	1.000	38.0	27.3	31.8	39.5	36.8	38.1
	36.8	53.3	43.6	89.6	81.0	85.1	49.2
	49.6						
	5.000	43.7	40.5	42.1	54.9	55.8	55.3
	42.1	46.4	44.1	91.3	87.2	89.2	57.3
	57.7						
	10.000	41.0	43.0	41.9	57.6	59.5	58.5
	42.9	42.8	42.9	92.3	85.0	88.5	57.5
	58.0	best f1	best Acc				
	20.000	40.0	41.2	40.6	54.9	59.5	57.1
	42.8	41.5	42.2	91.2	84.3	87.6	56.5
	56.9						
Confusion Matrix:							
	fan	gea	pump	val			
fan	10.70	3.57	10.08	0.56			
gea	4.94	14.10	4.08	0.56			
pump	8.69	5.60	11.22	0.68			
val	1.79	1.24	0.76	21.45			
Best macro f1 58.0							
Best macro f1 std 10.4							
Selected params: 10.00000							

Source: https://odin.tradoc.army.mil/WEG/Asset/755c0e281c53067c01a14af195f76eaf

ML Model: Support Vector Machine (SVM)

Support Vector Machine (SVM) is a powerful supervised learning model commonly used for classification and regression tasks. One of the key features of SVM is its ability to handle non-linear data through the use of kernel functions. The Radial Basis Function (RBF) kernel is one of the most popular and effective kernels used in SVM.





Questions?