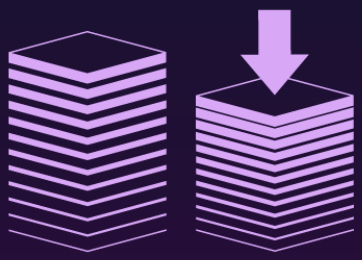


Lightscline

Data Reduction AI

Process the 10% important data from Terabytes of DAS data

August 2024
Lightscline, State College, PA, USA

About Lightscline

Lightscline's lightweight AI solves the foundational problem of efficiently analyzing Terabytes of data being generated from applications like urban infrastructure, traffic monitoring, oil & gas exploration, intrusion detection, construction, etc. Using Lightscline AI's 4 lines of code that can be setup within 10 minutes, customers can reduce 90% of their sensor data infra & human time and costs by selectively focusing on just 10% of the raw data. By exploiting the redundancy of real-world data, Lightscline AI makes real-time predictions using just 10% important data.

Executive Summary

In this whitepaper, we show how Lightscline's products can work with DFOS data, and how Lightscline AI can help in generating actionable insights using just 10% of the raw data. We analyze DAS datasets used to monitor subsurface environment changes, thunderquakes, traffic patterns, walk detection, and a music concert. We include (i) Data Labeling, (ii) Visualization, (iii) Model training approaches to understand how Lightscline's multi-channel analysis can be used to make the labelling & analysis significantly faster.

Applications Overview

Lightscline's lightweight AI unlocks several capabilities across space, aerial, terrestrial, and underwater applications which are currently unobtainable. Some applications include:

1. Quickly analyze 200+ hours of acoustic data for anomaly detection and ATR using just 10% important data
2. Order-of-magnitude reduction in the amount of training data without degradation to identification performance (Pid)
3. Prioritize training data by selectively focusing on the 10% important data
4. Enable transfer learning on the edge to quickly train for complementary tasks
5. Real-time inference on 15+ classes of data for on-board Human Activity Recognition (HAR)
6. On-board satellite-based AIS signal validation using 10% of the raw RF data

In this whitepaper, we will focus on DFOS data with applications like:

1. Construction detection for integrity monitoring in oil & gas applications
2. Walk / car / traffic detection using dark cables in urban environments
3. Natural phenomena and seismic activity monitoring – thunderquake detection
4. Detect micro-seismic events

Problem: Huge volumes of DFOS data

We start with a public DFOS dataset available here. The parent folder contains several sub-folder and files in tdms format, which contain the DFOS data collected over several months of experimentation.

The following figure shows the April 4, 2019 data files from a public DAS dataset in a state college. Data files are in. tdms format and can be sorted according to specific dates and times for visualization in a preferred python IDE.

Dataset size:
Rate of data generation = ~142MB per minute (with just 500Hz sampling frequency)
The size of the data for April alone is greater than 4.5 TB

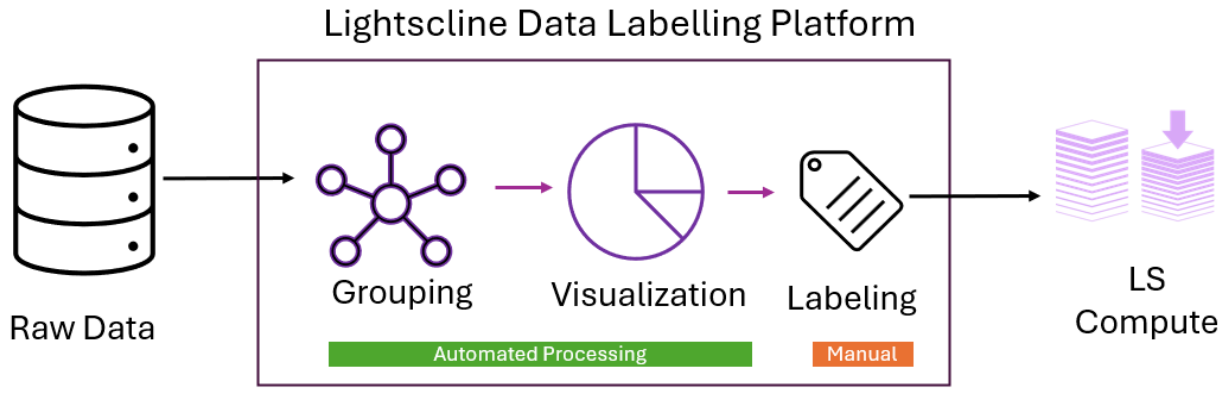
```
===================================================
Content of ZIP archive  apr-001.zip
===================================================

Archive:  apr-001.zip
  Length      Date    Time    Name
---------  ---------- -----   ----
        0  05-05-2021 20:30   apr/
149886976  08-22-2019 09:43   apr/PSUDAS_UTC_20190404_194804.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_194904.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_195004.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_195104.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_195204.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_195304.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_195404.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_195504.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_195604.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_195704.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_195804.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_195904.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_200004.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_200104.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_200204.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_200304.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_200404.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_200504.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_200604.509.tdms
149886976  08-22-2019 09:44   apr/PSUDAS_UTC_20190404_200704.509.tdms
```

This huge volume of data poses challenges in labeling, processing, and visualizing it. Moreover, ML models on this data require heavy compute hardware, which makes them difficult to train and deploy on the edge. Lightscline's Data Platform helps to quickly label and visualize the data. Lightscline's Edge AI helps in faster model training and reduces infrastructure costs. The next few sections will showcase how can Lightscline's products be used for this DFOS data.

Lightscline Data Platform

Lightscline Data Platform can help visualize and label the data. These labels can be used for Lightscline Compute or other machine learning frameworks for automatic inference from the data. This platform can also be used to find physics-based signatures in the data.

The Data Platform has visualization and grouping modules, which are described in detail.

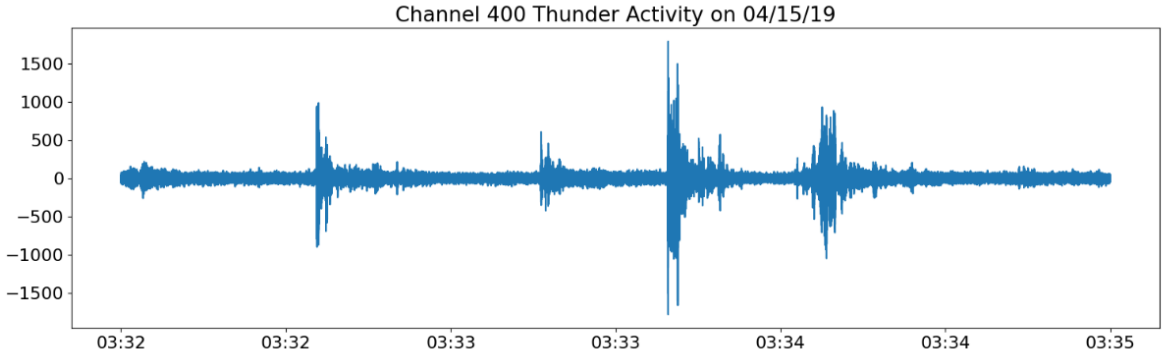


1.1 Data Visualization (& Use-case Definition):

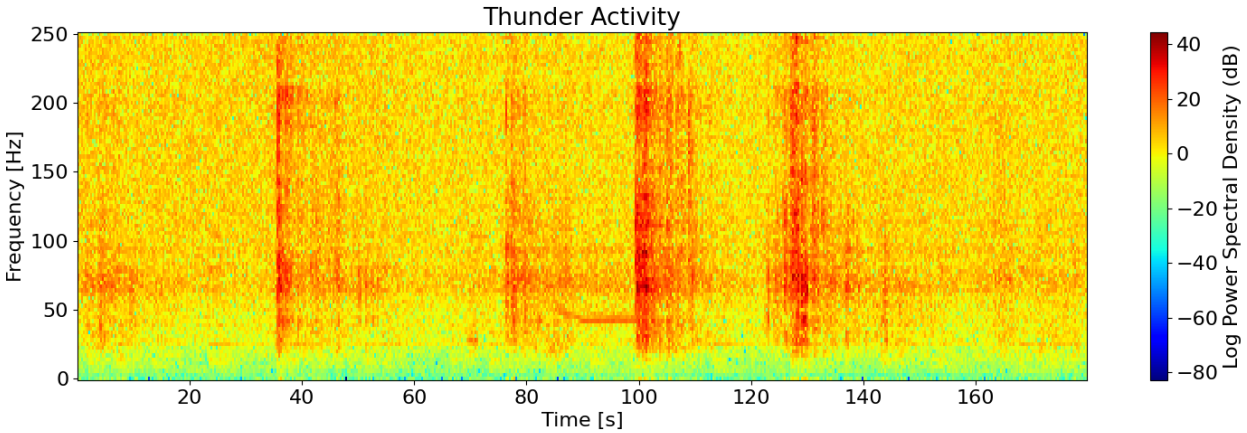
Our Data Platform provides visualizations of the data, which are useful in conventional approaches. These visualizations can be used to define which signatures are important and needs to be labeled. The goal is to automatically detect those signatures in the data after the deployment.

In the examples below, the data from DAS network in State College is visualized for multiple signatures. The dataset is public and can be found here. Though the examples shown in the figures contains only urban activities, similar signatures are observed for applications in infrastructure monitoring, oil & gas exploration, intrusion detection, etc. Data visualization for single channel, multiple channels, and looking at the spectrograms are the initial pre-processing steps for any DAS related data analytics.

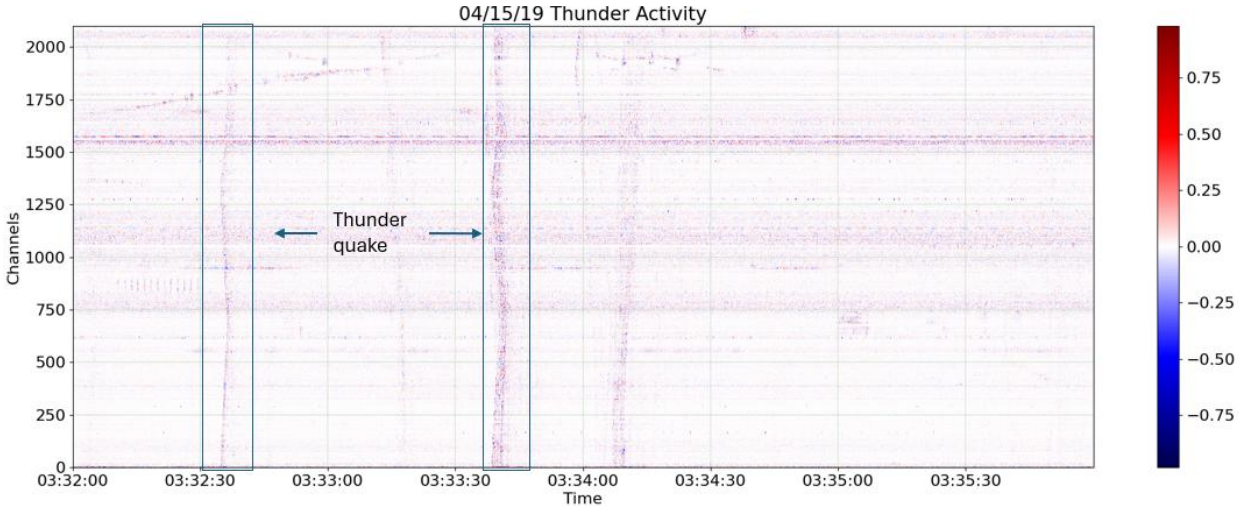
Thunderstorm activity: The following is an example of data visualizations that may be needed to detect thunderstorms using the DAS Network. The audio file generated from DAS data can be found here. [You will have to download the file to listen]



Spikes in single channel due to thunderquake.

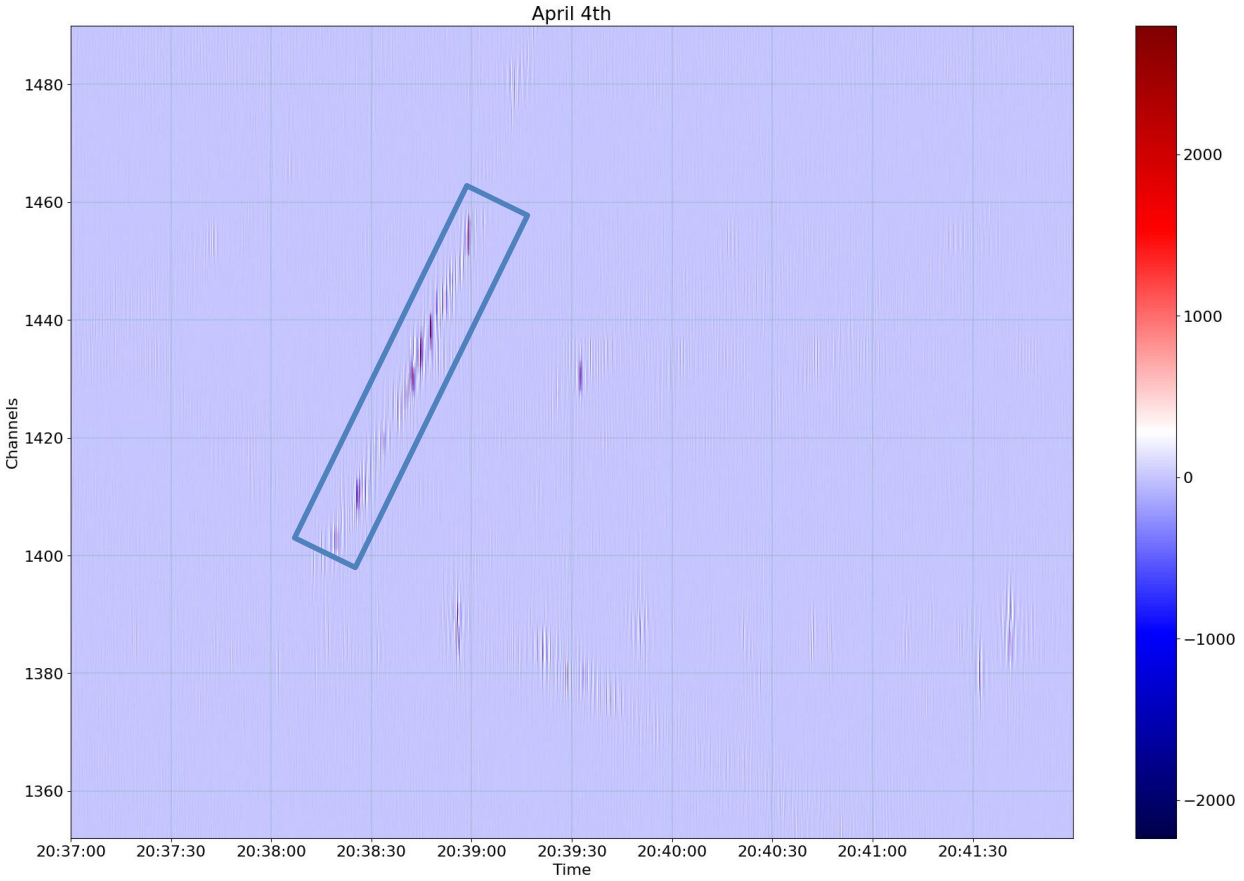


Spectrogram of a single channel. Frequencies 10-150 Hz are excited the most in a Thunderstorm



Thunderquakes are highlighted during a thunderstorm.

Footstep detection:



Channels are set up in the downtown area of State College, PA, a college town with a relatively low traffic footprint.

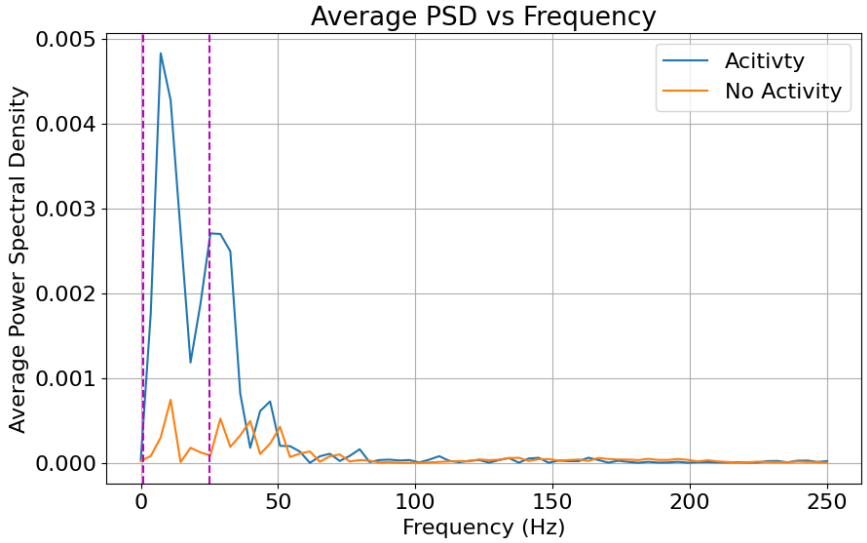
Based on the feature seen in the above plot and the information about our DFOS pipeline, we can make the following inference:

Channels spacing = 2m
Activity can be seen for ~65s
Activity seen from channels = 1460-1400 = 60
Velocity = 120m/65s = 1.85mph
The detected activity is walking.

Now all the similar data can be grouped automatically and labeled “Footstep Activity”. This significantly reduces the labeling time, as you only need to label one activity once.

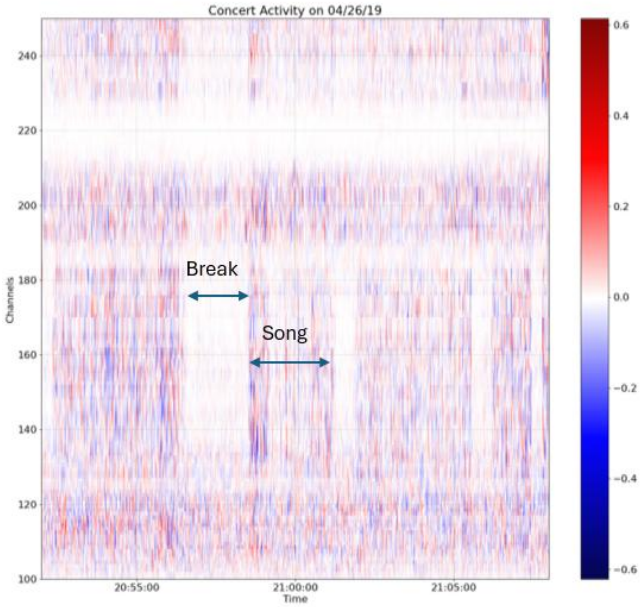
The following example shows no activity during 12:30-12:35 AM. This is expected downtowns are quieter at night during week days.

All of this data can be grouped together and labeled as “No Activity” to create labeled data.

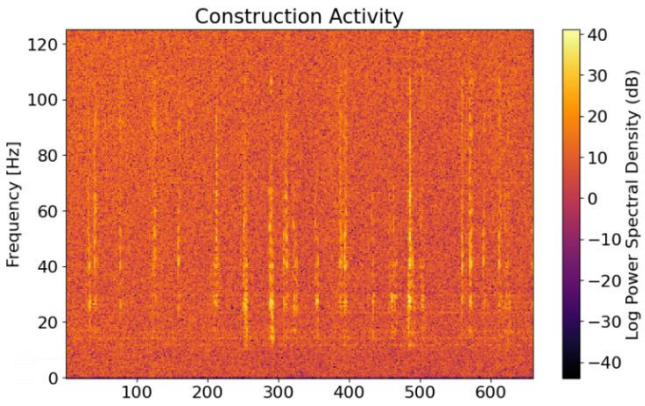


In this graph, the average PSD is much stronger in frequencies 1-25Hz. These frequency ranges are generally where footsteps and vehicles are detected. Different visualizations like this can be viewed to ensure that the activity is identified correctly.

The same dataset can be used to visualize and label different phenomena like construction, musical concerts, thunderquakes. Etc. These visualizations can include data from specific channels, spectrogram of one channel, PSD, change in Energy across time etc.
Here is an example of these phenomena Each of these activities can be added as another label.



Data showing the songs playing during Penn State's Movinon Concert.



Construction activity was detected on the western campus. The spectrogram of this activity was used to label the window as construction

1.2 Data Grouping:

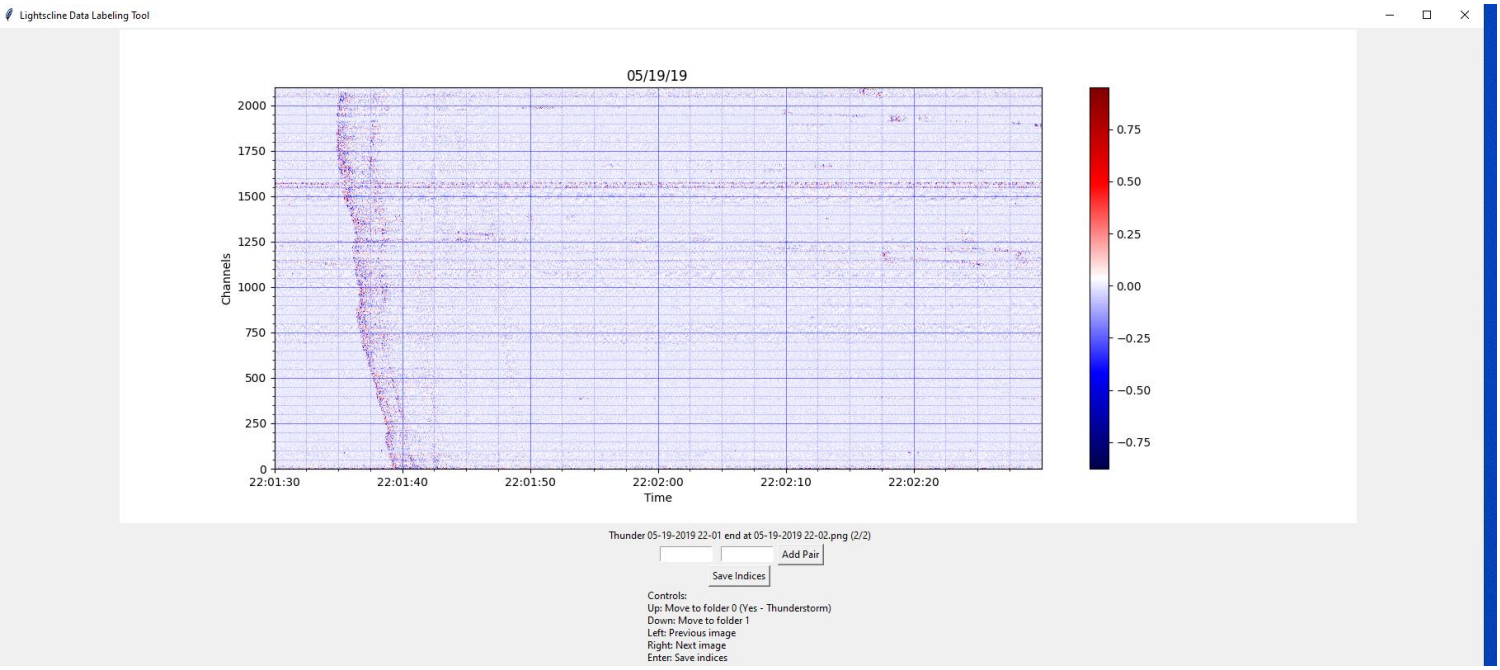
Lightscline Data Platform groups the data before humans start labeling them. Lightscline Data Platform also has fast visualizations for the grouped data. If one group member is manually labeled, all members will also be automatically labeled. This enabled a Fortune 500 company to manually label only 100 windows instead of the original 10 million windows – a 100,000x more efficient labeling process! Grouping similar data saves time & costs and ensures high-quality labeled data.

1.3 Labeling:

Loading from disk and plotting/rendering them on screen is slow and could take a few minutes per image. Especially because sometimes, the data had to be brought to RAM from the local disk, which might have to get it from the local network. Therefore, one or more visualizations (Spectrograms, single- and multi-channel plots) of each chunk (window) of data are made and stored for retrieval. This is done so that visualizations can be quickly retrieved during manual labeling process. Since the process of creating data visualization might be slow, it is decoupled from Manual labelling. It is parallelized and scalable across multiple machines, ensuring swift and seamless handling of large datasets.

Our intuitive labeling software features a streamlined user interface that allows for efficient keyboard-driven interactions. This approach has significantly accelerated the labeling process, enabling the annotation of **500 minutes of data in just 10 minutes.** Here is a video explanation of the UI where we explain the UI and label 60 min of data live.

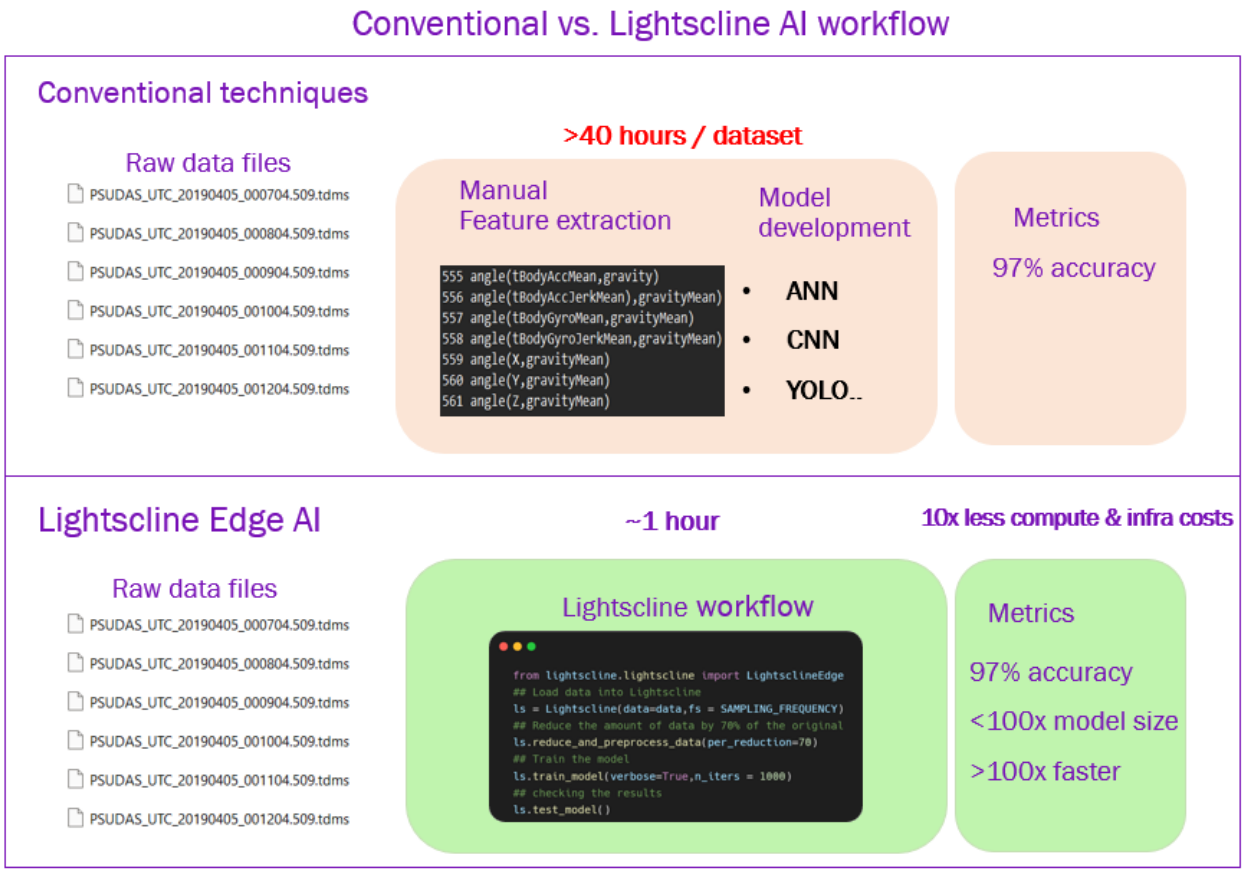
Here's an example of visualizing and labeling data using Lightscline's labeling software:



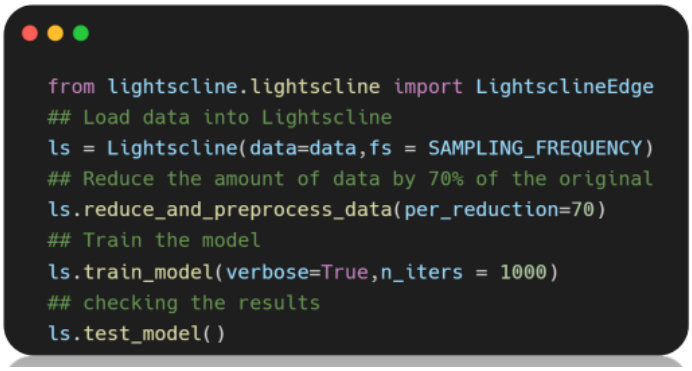
Lightscline Compute AI

After data labeling, the typical next steps involve either manual feature extraction or training the model on the full raw dataset. While feature extraction can result in smaller models, it is time-consuming because data scientists must manually select the features to be extracted for each dataset. On the other hand, training on the full raw dataset can achieve better accuracy but leads to significantly larger model sizes. By leveraging the redundance of real-world sensor data, Lightscline Compute automatically identifies a small fraction of important data for different downstream tasks. This approach eliminates the need for manual feature extraction and results in much smaller models, saving both time and costs.

The diagram below illustrates how Lightscline Edge achieves these efficiencies.



Lightscline AI



```
from lightscline.lightscline import LightsclineEdge
## Load data into Lightscline
ls = Lightscline(data=data,fs = SAMPLING_FREQUENCY)
## Reduce the amount of data by 70% of the original
ls.reduce_and_preprocess_data(per_reduction=70)
## Train the model
ls.train_model(verbose=True,n_iters = 1000)
## checking the results
ls.test_model()
```

- 4 lines of code to get started
- Setup within 10 mins
- No data sharing required

Several Fortune 500 companies have used Lightscline Edge AI to see the benefits of Lightscline Edge themselves. Users can get started with Lightscline AI using just 4 lines of code that can be set up within 10 minutes without any need for external data sharing. This product can be run on the customer's cloud environment or on-prem. It can also handle multi-modal data from different sensors at different sampling frequencies.

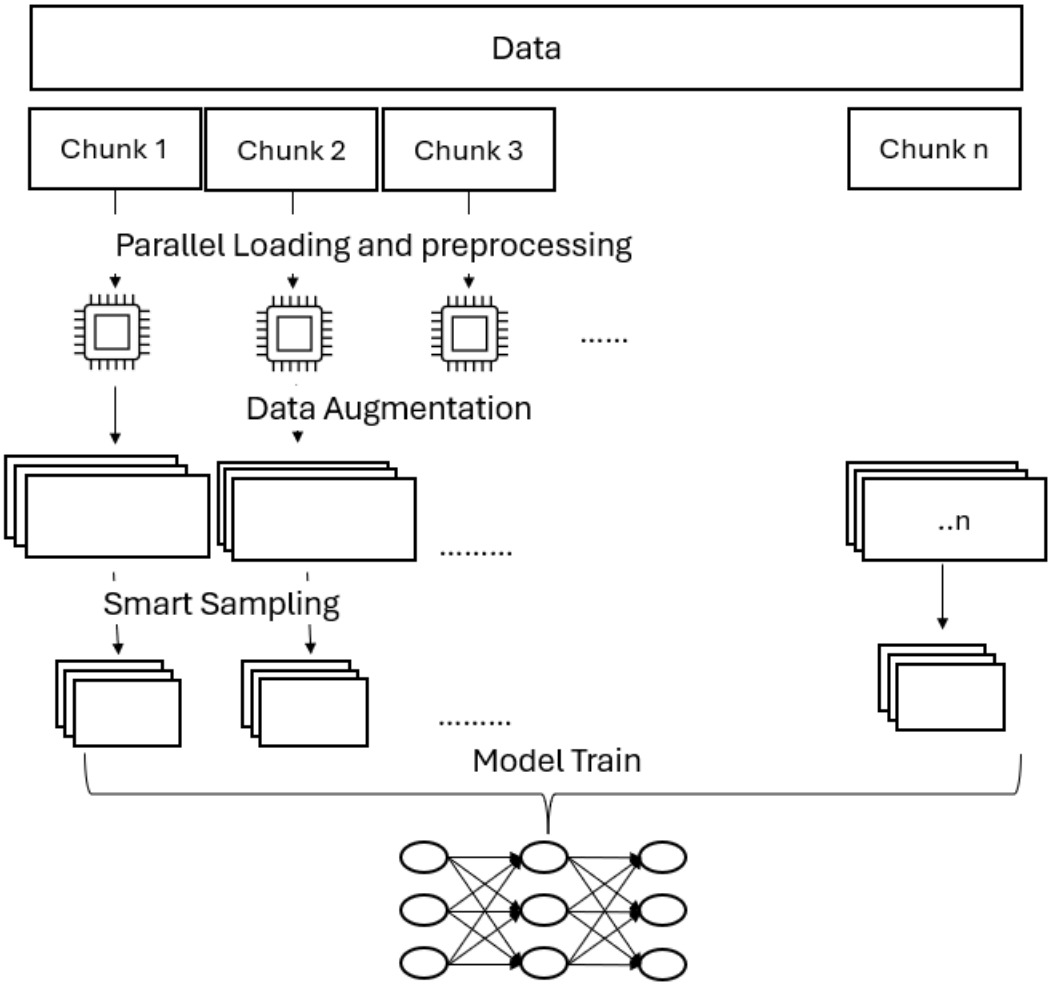
Here are the challenges involved in dealing with DFOS data:

1. Data Ingestion: A key challenge with DFOS data is its immense size, which makes it impractical to load the full terabytes of data into RAM. To address this, we load only small batches of data at a time. The data loading process is parallelized to optimize efficiency. While the model trains on the current batch, the next batch is preloaded in the background, minimizing idle time and maximizing training efficiency.

2. Data Augmentation: We leveraged data augmentation to train the model using only **17 original events** that happened during an hour. We augmented the set of 16 events into over 400+ events. This enriched data was then used to train the model effectively.

3. Smart Sampling: We utilized only 5% of the data from each chunk/window to train the models, with these data points intelligently selected in real-time using our proprietary algorithms. By training the model on just 5% of the data per window, we achieved a smaller model size, resulting in more efficient architecture. This approach also allowed us to fit more data into RAM, further optimizing the process.

The overall workflow is shown below:



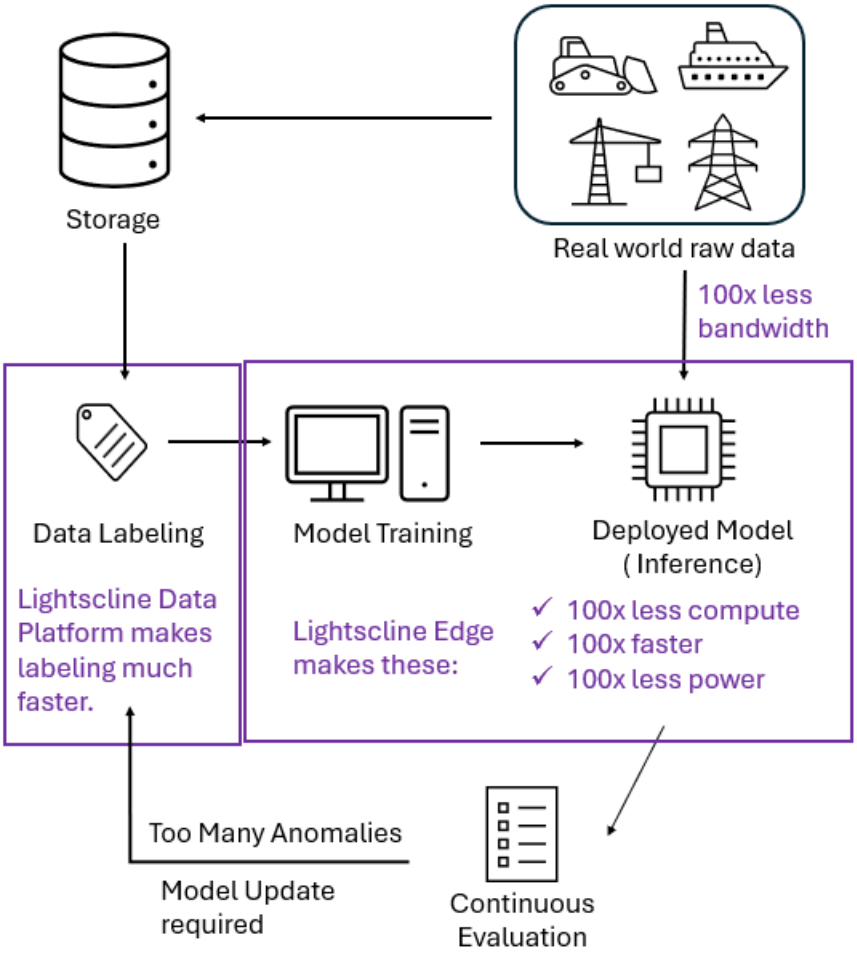
Workflow for dealing with large datasets

We were able to analyze **25 mins of 2100 channel data (~4km) per second**. The models can be trained the model was less than 5 minutes. We observed that disk I/O operations were a bottleneck during training. To address this, we implemented multiple levels of caching to improve performance. All these optimizations are turned on by default and can be controlled easily through software. Here is a video showcasing the prediction of data in real time.

Cick here to book a demo for this product.

Machine Learning CI/CD Pipeline

Lightscline Edge optimizes the entire CI/CD pipeline for model deployment, making it more efficient, faster, and cost-effective. By leveraging a small percentage of raw data for training, Lightscline Edge generates smaller models, which in turn reduces training and retraining time, and leads to significant savings in infrastructure costs.



Conclusion

Lightscline AI performs real-time data collection through prediction with >10x speed and energy efficiency over conventional approaches governed by the Shannon-Nyquist sampling theorem. This leads to orders of magnitude savings in (i) data infrastructure costs and time, and (ii) human resource efficiency. Additionally, this enables several new applications not possible today due to extreme SWaP-C requirements of real-world AI applications.

Reach out to info@lightscline.com for any queries.