

TRAIN THE NEURAL NETWORK

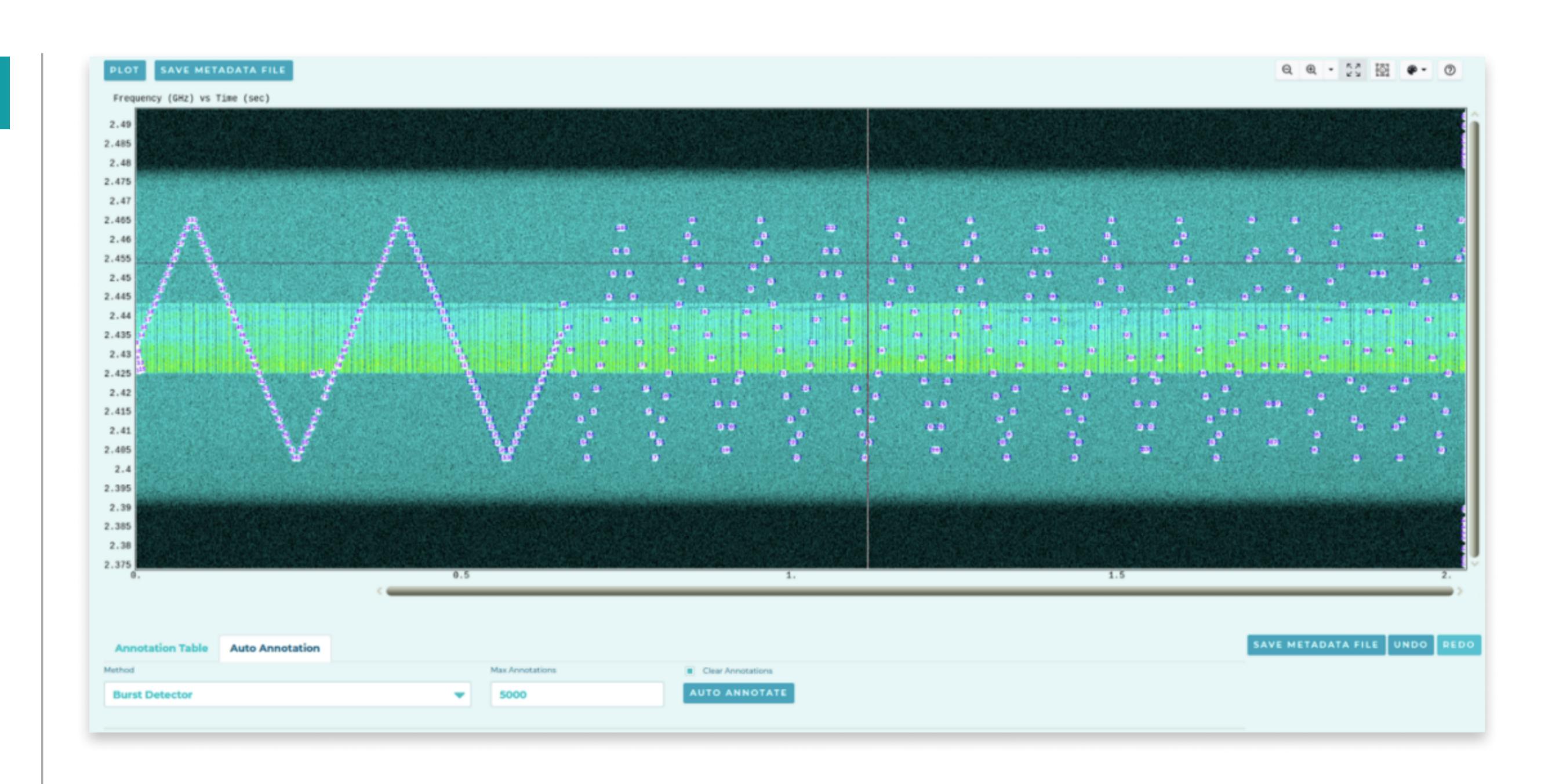
OmniSIG® Studio

DATA-CENTRIC AI

- DeepSig has created and maintains a world-class labeled dataset of over-the-air and synthetic RF signals.
- The pre-made dataset allows you to start with terabytes worth of labeled data, resulting in significant time savings in labelling data and increased accuracy results for new models.
- Users can complement the DeepSig dataset with additional signals, even if the DeepSig dataset already contains samples of that signal type.

NO-CODE

- Studio's Graphic User Interface alleviates the need to write code when labeling data or training the neural network.
- This innovation significantly increases the number of users who can rapidly create new RF signal detection models.



OmniSIG Studio puts the power of Artificial Intelligence in your hands by allowing you to train OmniSIG's signal detection & classification neural network with an easy-to-use user interface, no-code, online/offline, software application. Create new signal detection and classification models, label raw radio signal recordings and allow the convolutional neural network to train and learn new features dramatically faster than traditional methods.

CUSTOMIZE SIGNALS

Create a custom, Al-enabled, RF detection & classification system for your specific needs, with no prior Al development experience needed. Load RF snapshots for signals of interest, use drag and drop methods to label and annotate custom signals & use your new training data to evolve a neural network. This allows for new capability creation without writing any Digital Signal Processing (DSP) code.

ADVANCED AI MADE EASY

As you interact with Studio, a convolutional neural network is molding itself with the knowledge you front-end, you can load RF captures and present it. Hundreds of layers, with thousands of parameters, are constantly adapting to the new input, resulting in an optimal solution for your specific requirements.

FAST AND EASY DEPLOYMENT

Every Studio includes a single OmniSIG Software license, providing you with a fully customizable testing environment. Once satisfied with the performance of the RF classification model, you can quickly load a deployable OmniSIG with the new model, enabling a fast turnaround for scenarios requiring solutions to be developed, tested, and deployed.

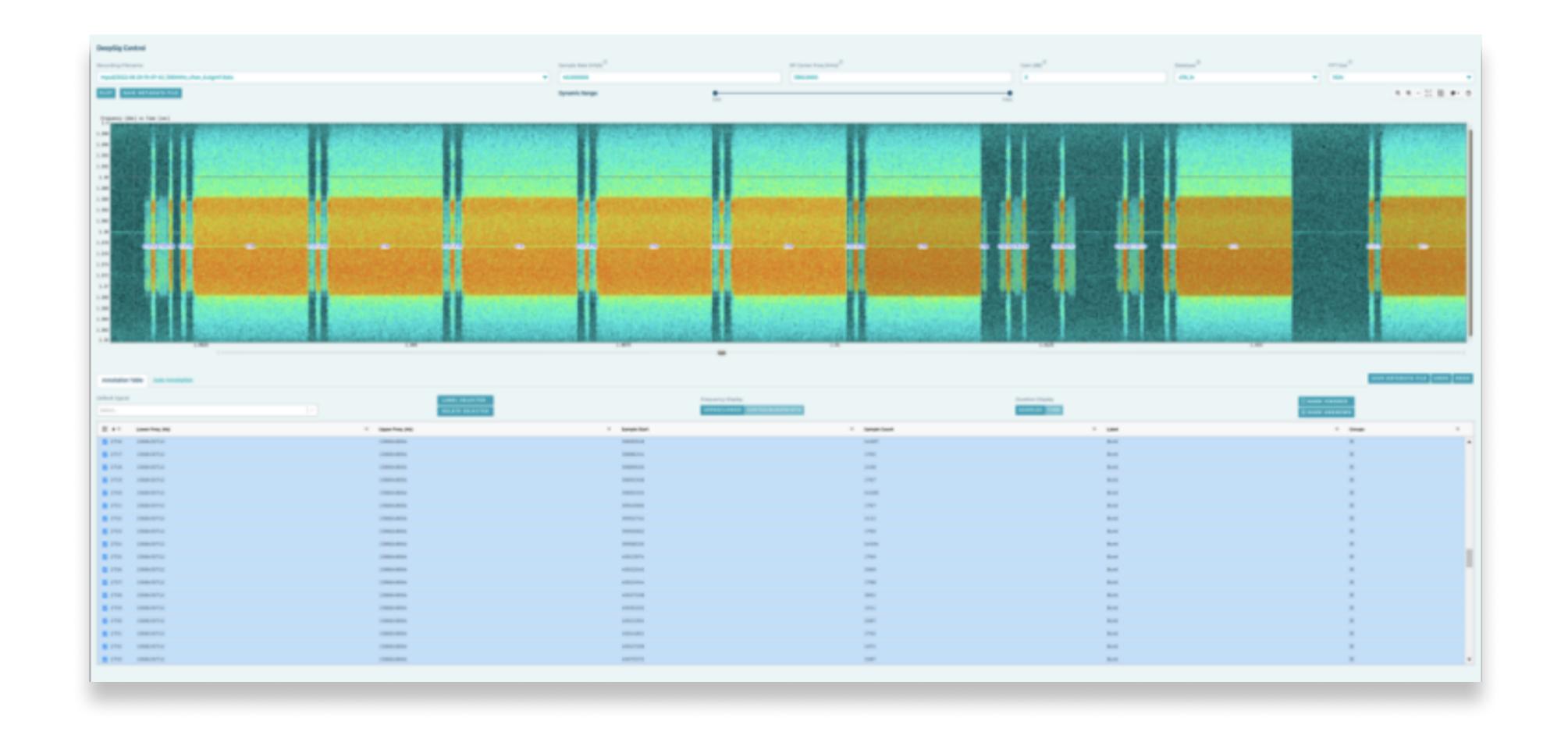
WEB-BASED INTERFACE

Packaged with an easy-to-use web-based simultaneously interact with the spectrogram, simplifying the data labeling process. After the labelling and annotation is complete, use Studio's training page to start, monitor, & quickly test new OmniSIG neural network models.

LABEL AND ANNOTATE DATA WITH EASE

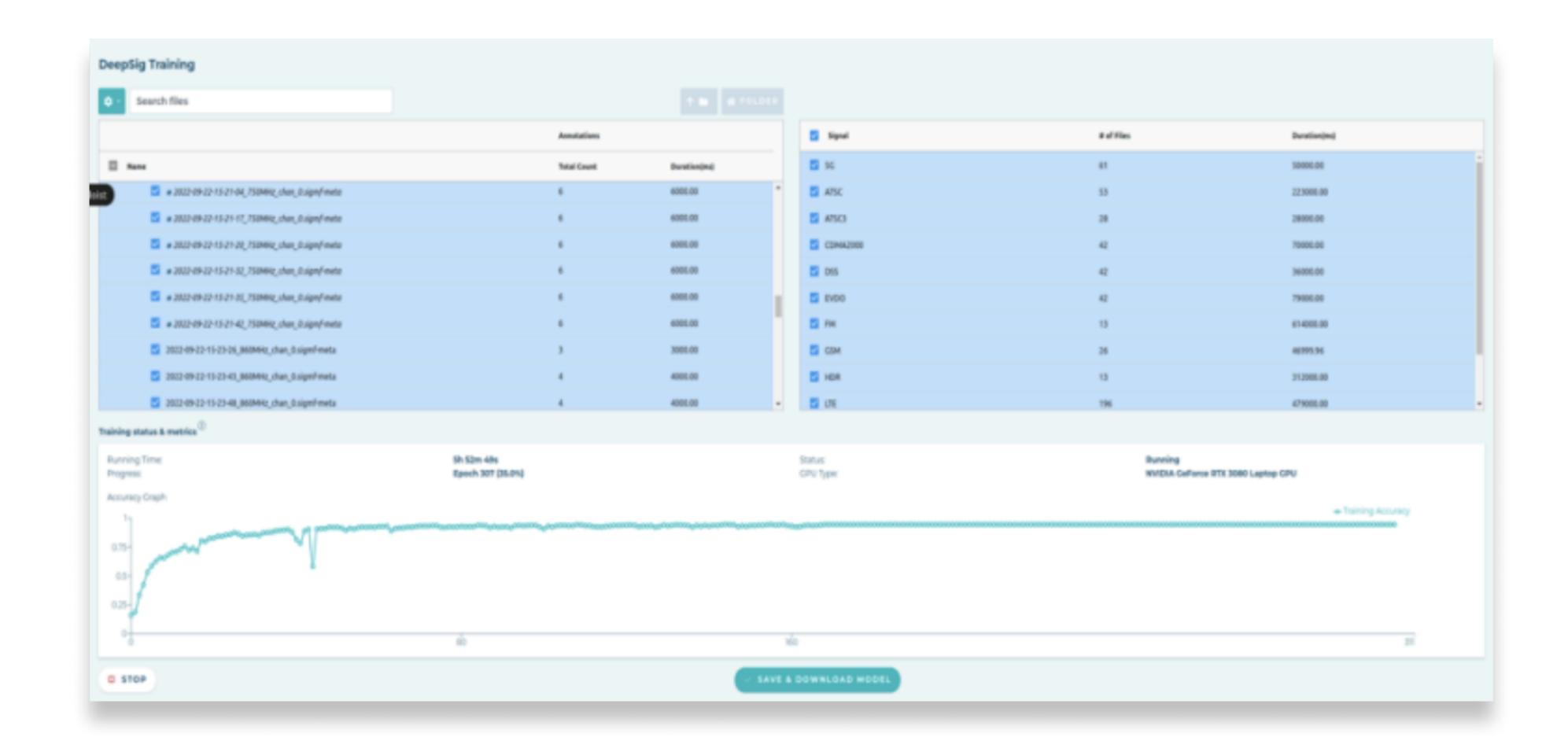
The performance of a ML-based signal classifier is dependent on the dataset used to train the model. DeepSig engineers have decades of combined RF experience and have spent years building world-class RF datasets. DeepSig has openly published several datasets to foster increased academic focus on leveraging AI for RF applications. The OmniSIG model is trained on data collected across the globe, enhancing OmniSIG's ability to have high performance detections and classifications worldwide.

OMNISIG STUDIO USER INTERFACE



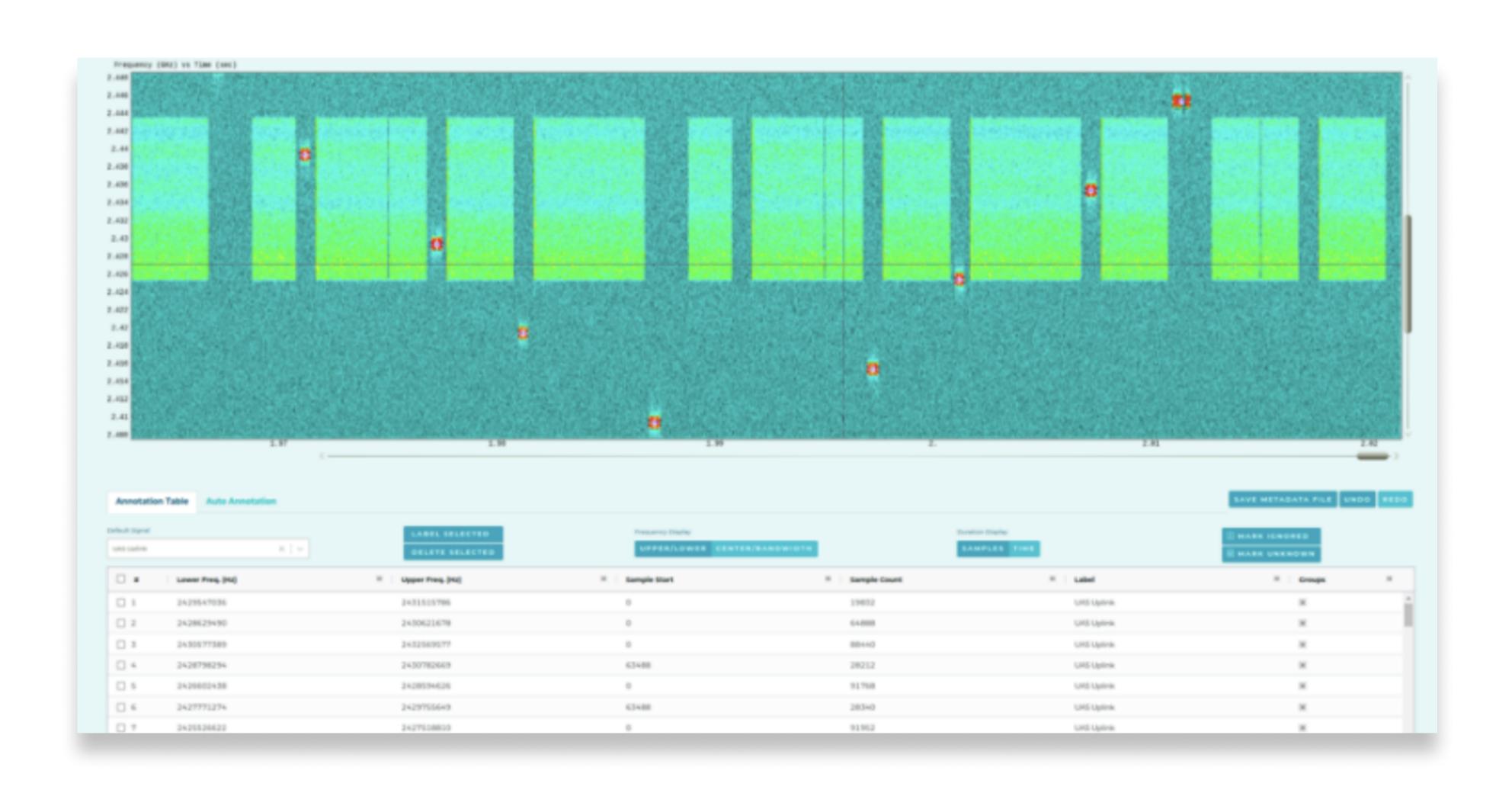
SELECTIONS

The labeling tool keeps track of all annotated signals and displays them to users in the signal selections table. Quickly traverse RF snapshots for fast & easy data labeling. The tools are also deployable as remote services to enable disjoint users to work together on labeling data.



TRAINING EXECUTION – MONITORING – DEPLOYMENT

Monitor the training status using the metrics windows on the training tab. By adding data, you can control the performance of your model and save for deployment at any time.



AUTO-ANNOTATION

Label signals quickly and clearly with Studios' annotation tool. You are now able to label thousands of signals in seconds and have a display of all annotations and metadata in an easy-to-read table. The figure shows the results of the auto-annotation tool when labeling the uplink bursts from a drone link.

deepsig.ai