



R.I.C.E

February 1, 2019

Jacob Fletcher, Emil Mathew,
Matthew Sumpter

Overview -R.I.C.E(name in development)

- Radio frequency
 - Interception
 - Classification
 - Exploitation
-
- Small operator carried system, that can work with long term missions and shorter DA operations.

Background – Why?

- Environments where current EW capabilities are limited
 - Dense Urban Environments
 - Subterranean Combat Environments
 - Operations where Near Peer A2/AD capabilities that limit the deployment air/ground assets
- Increasing global usage of IoT and RF main medium of wireless communication
 - 5.4 billion IoT
 - ~5 billion cellphones

Background – Why?

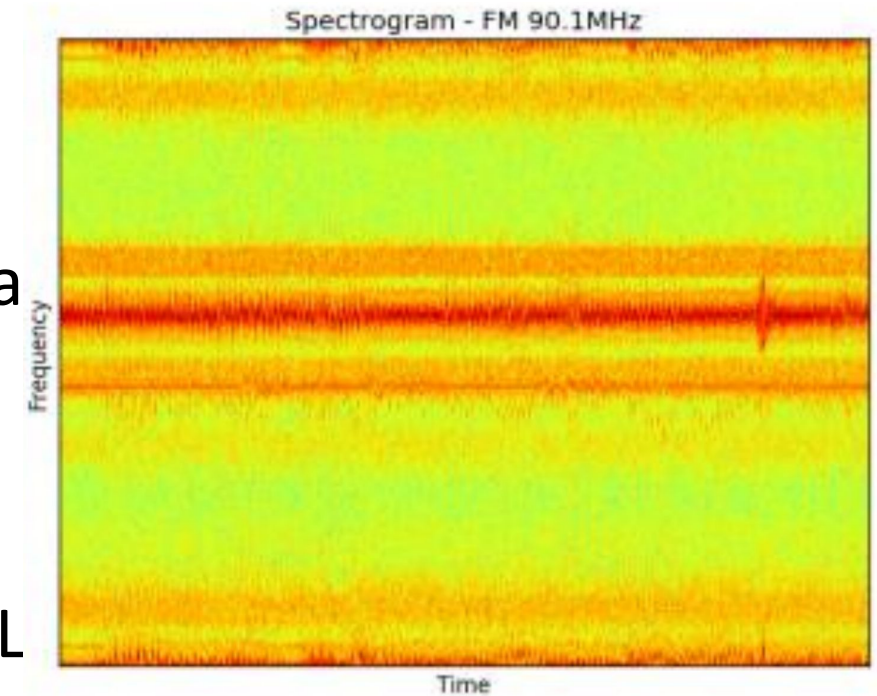
- In dense urban areas, signal traffic is high and difficult to identify
 - RF “Forensics”
 - What signals are present?
 - What signals are important?
 - What signals aren’t following the rules?
 - Security concern – may be intended to hack or “spoof” devices

Background – Why Machine Learning?

- RF sampling rates can exceed 200 GB/s
- Machine Learning:
 - Eliminates human error
 - Proper signals captured
 - AND storage of interesting data

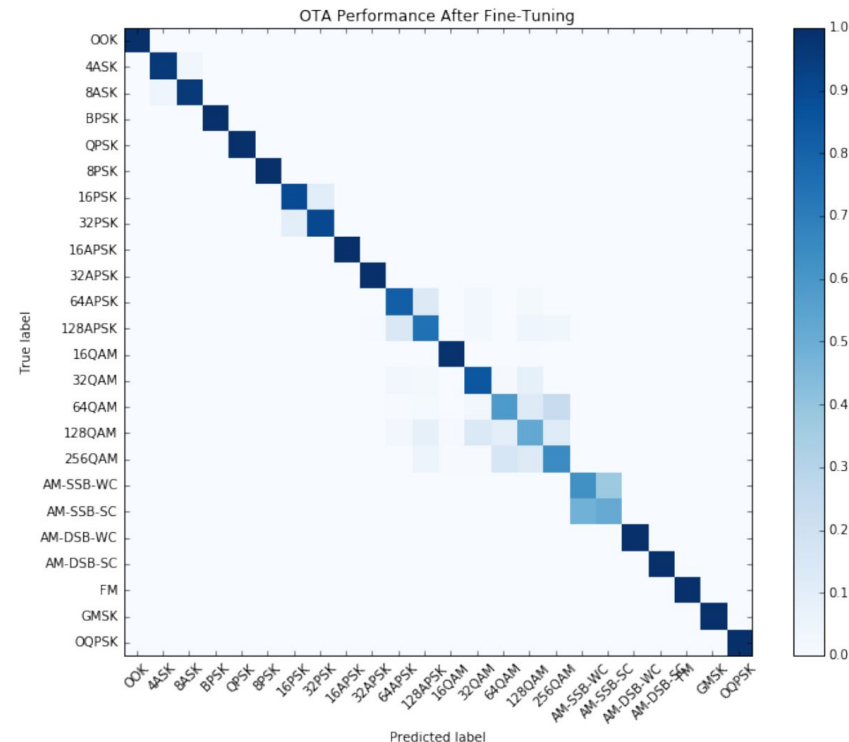
Background – Previous Work

- Spectrogram Approach
 - Discards phase info
 - Can be ~99% success at signal identification with expensive and heavy equipment
 - Pros:
 - Utilizes well-established tools for ML
 - Effective at signal identification
 - Cons:
 - Extra level of processing required to generate image



Background – In-phase/quadrature (IQ) Domain

- Uses DeepSig generated dataset and Convoluted Neural Network
- Pro: Facilitates real time decisions ~94/87% success (simulated/OTA)

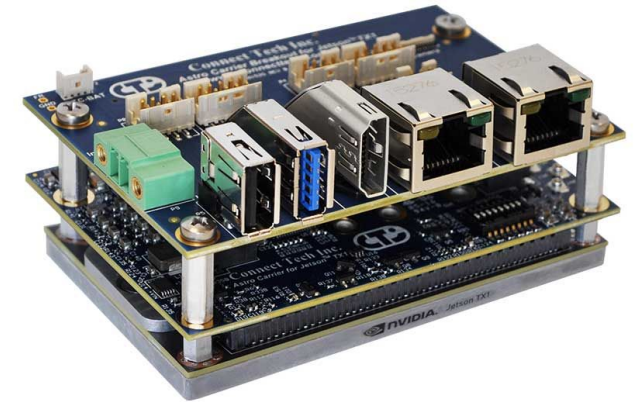


Background – Challenges

- Preprocessing and machine learning on data of this complexity not well explored in non-proprietary sources
- Practical application:
 - Classifying RF in real-time proven successful
 - Can it be made portable?

Tools for Approach – Hardware

- Software-Defined Radio Devices
 - RTL-SDR
 - LimeSDR/Mini
- TX2
 - Portability allows for easy carry by operators
 - Capable of computational demands

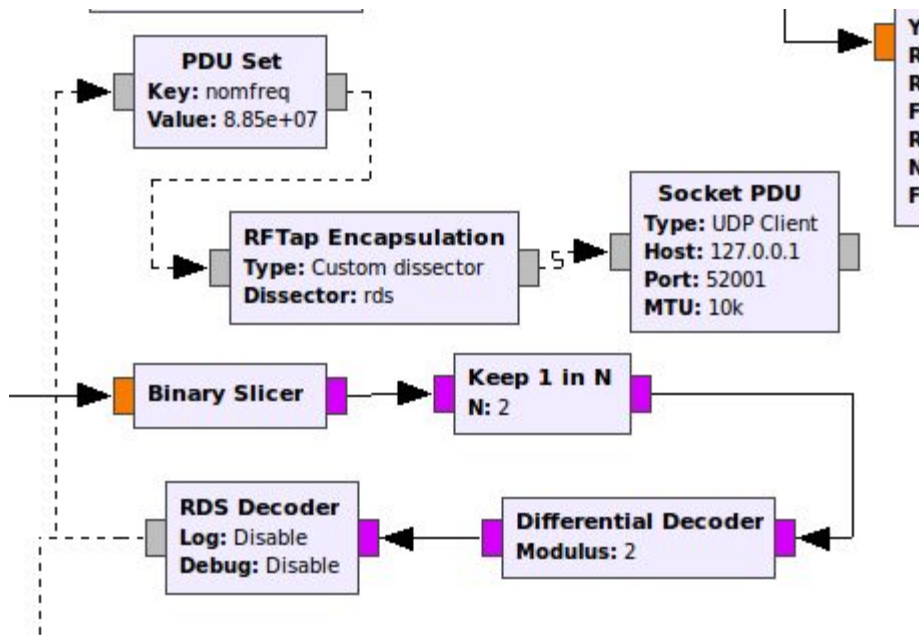


Tools For Approach – Software

- GNURadio
- RFTAP
- Wireshark



Current Interception Format



The screenshot shows a Wireshark capture of RDS data. The top pane displays a list of packets, and the bottom pane shows the details of the selected packet (Frame 81).

No.	Time	Source	Destination	Protocol	Length	Info
73	3.148767772			RDS	78	PI=D393 GRP=3A
75	3.270853421			RDS	78	PI=D393 GRP=4A
77	3.295081168			RDS	78	PI=D393 GRP=8A
79	3.398178880			RDS	78	PI=D393 GRP=11A
81	3.526447006			RDS	78	PI=D393 GRP=0A <WD> AF=89.8MHz
83	3.544453316			RDS	78	PI=D393 GRP=0A <R> AF=89.8MHz
85	3.664056311			RDS	78	PI=D393 GRP=0A <3> AF=89.8MHz
87	3.793449711			RDS	78	PI=D393 GRP=0A <> AF=89.8MHz
89	3.912973802			RDS	78	PI=D393 GRP=1A
91	3.927027210			RDS	78	PI=D393 GRP=2A
93	4.052358448			RDS	78	PI=D393 GRP=2A

Frame 81: 78 bytes on wire (624 bits), 78 bytes captured (624 bits) on interface 0
Ethernet II, Src: 00:00:00:00:00:00, Dst: 00:00:00:00:00:00
Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
User Datagram Protocol, Src Port: 39877, Dst Port: 52001
RFTap Protocol (24 bytes)
RFTap Fixed header
Nominal Frequency: 88500000.000000 Hz
Dissector Name: rds
Radio Data System (RDS)
PI code: 0xd393
0000 = Group type code: 0
.... 0... = Version code: 0
AF1 code: 225
AF2 code: 23
Alternate Frequency: 89.8 MHz

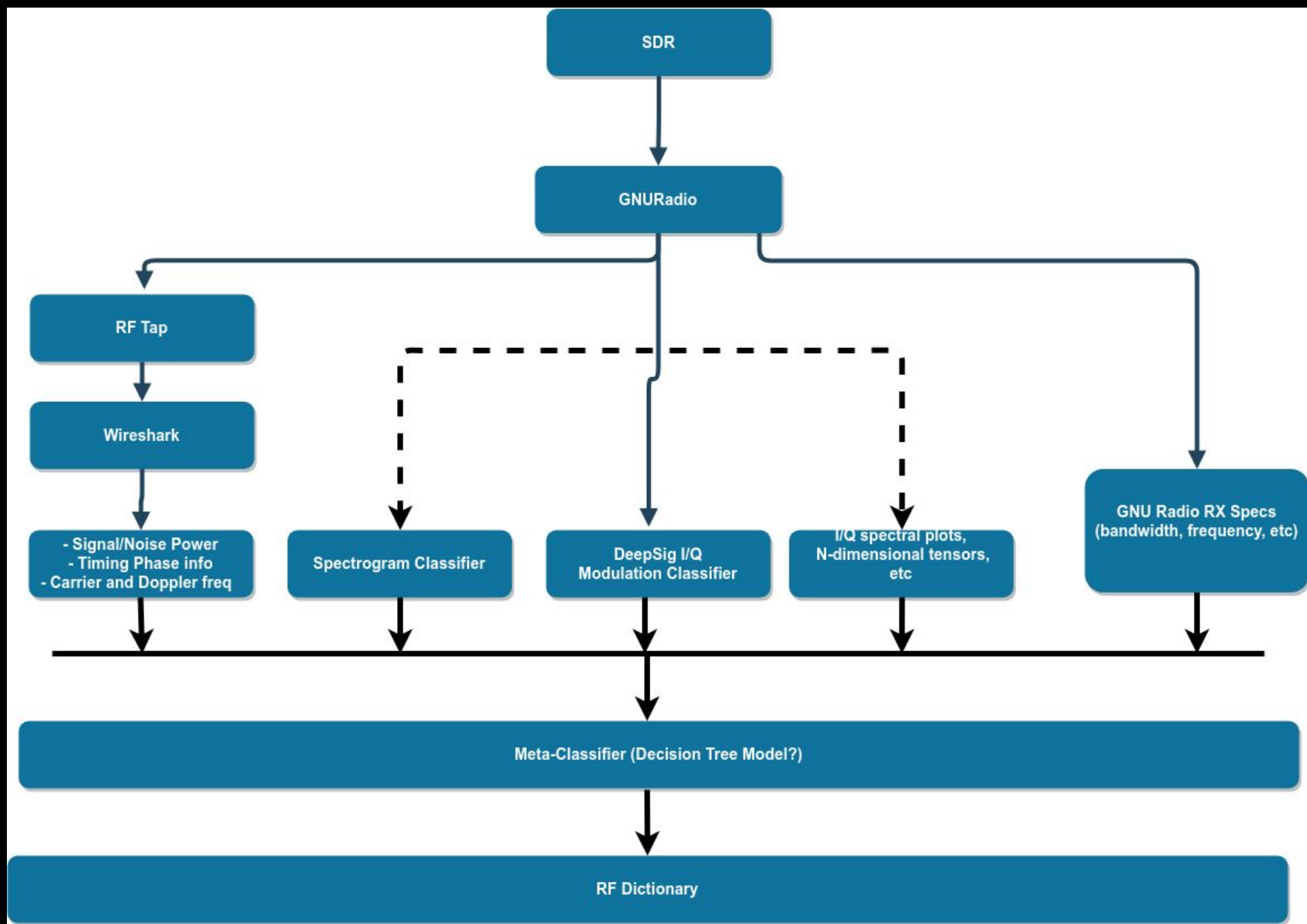
0000 00 00 00 00 00 00 00 00 00 00 00 00 08 00 45 00E.
0010 00 40 9f 69 40 00 40 11 9d 41 7f 00 00 01 7f 00 .@.i@.@.A.....
0020 00 01 9b c5 cb 21 00 2c fe 3f 52 46 74 61 06 00!.,?RfTa..
0030 04 00 00 00 00 80 9c 19 95 41 10 00 03 ff 72 64A....rd
0040 73 00 d3 93 05 c8 e1 17 57 44 41 42 43 44 s.....WDABCD

Generating Data – Processing RF signals

- GNU Radio config. provides initial insight
- GNU Radio Modules
 - RFTap
 - Allows wireshark to view RDS packet headers and look for metadata flags
 - Data “grooming”
 - Trim focus to specific bandwidth/frequency/modulation
 - Eliminate noise

Classification – Ensemble Classification

- I/Q currently best standard for real-time analysis
- Stacking
 - Weight predictions from various classifiers
 - Design meta-classifier that determines final classification from all garnered meta data



Future Work – RF Dictionary

- There is still a need for an open, collected dataset
- Incoming signals can be quickly identified and assessed
- New signals can be incorporated
- SQLite DB with metadata used to build reference library

Future Work

- Build multiple schemas for detection
 - Automate movement between schemas with python
- Explore other classification methods
- Need for transfer learning
 - Training is expensive
 - Classify real-time data, feed back into classifier training model
- Focus on UHF and VHF for interception and recording

Resources

<https://www.darpa.mil/news-events/2017-08-11a>

https://www.thinkmind.org/download.php?articleid=data_analytics_2015_8_30_60114

<http://on-demand.gputechconf.com/gtc/2018/presentation/s8826-deep-learning-applications-for-radio-frequency-rf-data.pdf>

<https://arxiv.org/pdf/1712.04578.pdf>

<https://blog.kickview.com/deep-learning-meets-dsp-ofdm-signal-detection/>