Data Analytics for Wireless Communication and sensing

Lili Qiu

UT Austin
Use Cases

- Predict wireless performance [INFOCOM’16]
- Network test site selection and diagnosis [ICNP’17]
- Wireless Sensing [MobiCom’19]
Use Cases

• Predict wireless performance [INFOCOM’16]
• Network test site selection and diagnosis [ICNP’17]
• Wireless Sensing [MobiCom’19]
Limitation of Average SNR

- Avg. SNR is widely used for delivery ratio estimation

SNR is not a good predictor in Freq. Selective Channels

Ref. Halperin10
Frequency Diversity

- Wireless channel are frequency selective
- Bandwidth is increasing with new standards

Due to frequency diversity, average SNR cannot predict wireless performance.
Effective SNR

• Effective SNR shows better results than Avg. SNR

• Approach
  1. Map the SNR per subcarrier to BER

\[ BER_{eff,k} = \frac{1}{N_{subs}} \sum BER_k(SNR_k) \]

  2. Map \( BER_{eff,k} \) back to Effective SNR

\[ \rho_{eff,k} = BER_k^{-1}(BER_{eff,k}) \]

  3. Use Effective SNR to select the appropriate rate

How Accurate is Effective SNR?
Effective SNR Accuracy

- Scatter plot of Estimated delivery ratio vs. gnd. truth

Effective SNR is also not very accurate
Error Burstiness across Frame

- Assumes that interleaver uniformly distributes bit errors
- WiFi interleaver has a skewed distribution
- Error still bursty in WiFi interleaver

Delivery rate estimation must incorporate error burstiness
Problem Formulation

• Goal
  • To accurately predict delivery ratio using CSI information

• Options
  1. Analytical modeling
     • Hard for freq. selective channels.
  2. Simulate online
     • Prohibitively expensive
  3. Lookup table based approach
     • Pre compute delivery rate for error patterns
     • Error pattern capture the burstiness patterns
  4. Machine learning based approach
     • Use supervised learning to estimate delivery rate
Option 4. Machine Learning Approach

• Propose a machine learning based solution

• Motivation
  – Avoids the time and space complexity of lookup tables
  – Machine learning can provide faster online solution

• Machine learning algorithm
  – We chose Neural Networks

• Reason
  – Supports non-linear continuous functions
  – Appropriate for delivery ratio
Feature Set

• Feature Set
  • Use BER per subcarrier

• Advantage
  • Easy to obtain from CSI information
  • Allows de-coupling from the interleaver and modulation scheme
  • Feature size is limited by number of bits in OFDM symbol
Neural Network Operation

**Training**

- Input
  - Bit-Error-Rate
  - Delivery Ratio

- Intel Channel Traces
- TGn Channel models

- Neural Network

- Output
  - Network weights

**Testing**

- Input
  - Bit-Error-Rate

- Trained Network

- Output
  - Delivery Ratio
Delivery Ratio Accuracy (40MHz)

EffSNR
Wifi: 11%
Our: 27%

Lookup
Wifi: 4.5%
Our: 3%

ML
Wifi: 6%
Our: 3%
Use Cases

• Predict wireless performance [INFOCOM’16]
• Network test site selection and diagnosis [ICNP’17]
• Wireless Sensing [MobiCom’19]
Field Testing Process

Apply Upgrade (Trial)

Performance looks good. Go ahead with network-wide rollout

Network wide roll-out can have contrasting performance impact due to different location characteristics
Test Site Selection

• Problem: How to select test sites to maximize early problem detection?
  • Low sampling rate
  • Other contexts: car crash test, medicine design

• Bayesian experiment design

• Greedy heuristic: Incrementally choose locations that diversify feature values
  \[ \forall j, \quad \max_i \min_j (\text{Hamming}([n_i], n_j)) \]

• Testing 1% nodes can identify major features that affect upgrade performance
Use Cases

• Predict wireless performance [INFOCOM’16]
• Network test site selection and diagnosis [ICNP’17]
• Wireless motion sensing [MobiCom’19]
Signal Processing is Not Enough

\[ \mathbf{x} \text{ is the ground truth position} \]
Our Approach

- Received signals
- **RTrack**
  - Signal Proc.
  - RNN
- Target position

- Separate target reflection from others
- Exploit temporal relationship
Applying RNN

\[ \text{(d, } \theta) \]

Current profile

Recent profiles
Network Architecture

- Prev. context
- Dist & AoA
- Context info.
- Combine features
- Extract features
- Reduce the input size

2D profile

O
C
H2
H1
P
Context Layer

• Consist of 5 neurons

<table>
<thead>
<tr>
<th></th>
<th>$d$</th>
<th>$\theta$</th>
<th>$\nu_r$</th>
<th>$\nu_t$</th>
<th>$\alpha_r$</th>
<th>$\alpha_t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.94</td>
<td>0.04</td>
<td>0.06</td>
<td>0.00</td>
<td>0.34</td>
<td>0.09</td>
</tr>
<tr>
<td>2</td>
<td>0.20</td>
<td>0.76</td>
<td>0.14</td>
<td>0.06</td>
<td>0.08</td>
<td>0.30</td>
</tr>
<tr>
<td>3</td>
<td>0.03</td>
<td>0.24</td>
<td>0.79</td>
<td>0.13</td>
<td>0.28</td>
<td>0.10</td>
</tr>
<tr>
<td>4</td>
<td>0.10</td>
<td>0.31</td>
<td>0.31</td>
<td>0.22</td>
<td>0.03</td>
<td>0.07</td>
</tr>
<tr>
<td>5</td>
<td>0.01</td>
<td>0.10</td>
<td>0.04</td>
<td>0.06</td>
<td>0.06</td>
<td>0.07</td>
</tr>
</tbody>
</table>
Detection Window

- Use a local profile near the target

- Isolate environment impacts
- Save training efforts
- Support multiple users
- Reduce computation cost
Data Augmentation

• Generate more training data with existing data
• Our framework is easy to augment data
Thank you!

lili@cs.utexas.edu