CSE 891: Selected Topics - AIoT
Project 1 - Report

Neural Enhanced Decoding Model Reproduction and Enhancement

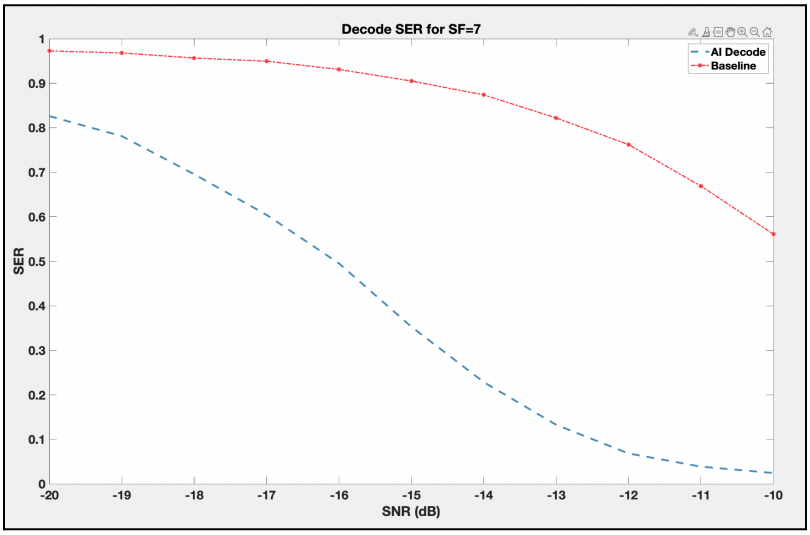
Rishika Juloori[1] , Riya Thakore[2] , Varun Kodandapuram[3]
[1]juloorir@msu.edu, [2]thakorer@msu.edu,[3]kodanda1@msu.edu

- **NELoRa reproduction and Dechirp process**

  - Build the dechirp model with MATLAB
    - The dechirp model implements the standard LoRa demodulation method while certain specific LoRa configurations for instance SF, BW are provided. This model determines an SNR threshold above which chirp symbols can be decoded.

  - Build the neural enhanced decoding model on your PC or server.
    - The limitation of the existing dechirp model has the root cause that the dechirp model only relies on the energy in the spectrum. This can be used as a motivation to redesign the LoRa demodulation method.
    - Authors presented NELoRa, which is a neural-enhanced demodulation method with the key idea to use Deep Neural Networks (DNN) to extract the fine-grained information for decoding purposes.

  - Compare the two methods
    - As the transition region of SNR is from -20dB to -10dB, in this project, we have only focused on this zone.
    - The two methods - Baseline dechirp performance and NELoRa can be compared as per the plot given below.



Plot 1.1

- **NELoRa Enhancement**

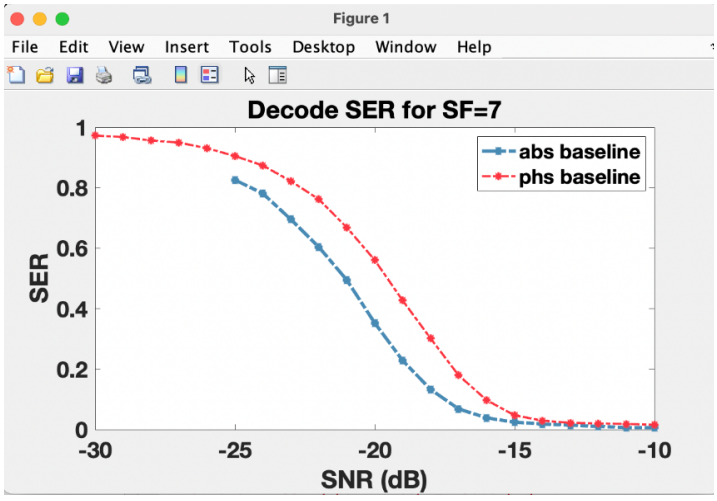
    - There are various methods that can be implemented in the model so that we get similar accuracy performance that takes less time and prune the network model to improve the SNR threshold and overall performance.

    - Since the NELoRa model is based on a neural network, we tried to hypertune the batch size and SNR list for both the baseline and AI decode performance to boost the accuracy.

    - Our main aim for this enhancement is to decrease the symbol error rate. Once the tradeoff between the batch size is estimated, we can claim that larger the batch size, lesser is the noise in the gradients and so better is the gradient estimate. This will help the model to reach minima early.

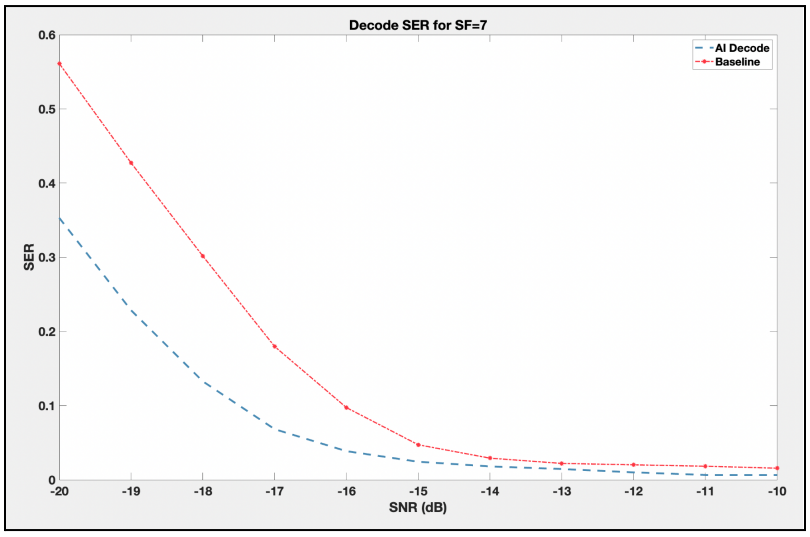
    - The lower the curve, the higher the accuracy and lower the error rate. Our enhanced model consists of the curve which is lower than the origin curve shown in plot 1.1. This implies that the error rate of the enhanced model is lower than the original model, shown in plot 1.3. Since accuracy and error rates are inversely proportional (lower error rate would imply higher accuracy), the lower rates in plot 1.3 imply higher accuracy.

    - Initially, we were getting the following plot plot 1.2, when the range was not as per the described one -20dB to -10dB, this served as an initial hypothesis for us to work on the modification. However in order to compare with the modifications of the model, we would compare our plot 1.3 with plot 1.1 that can clearly demarcate the differences between respective SER values.



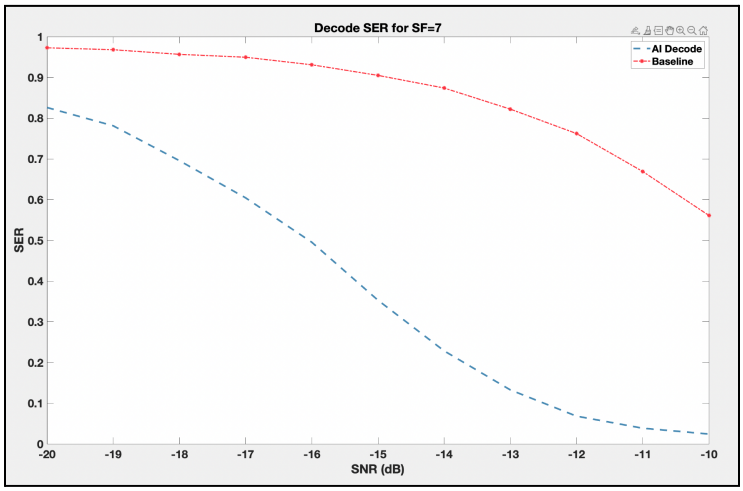
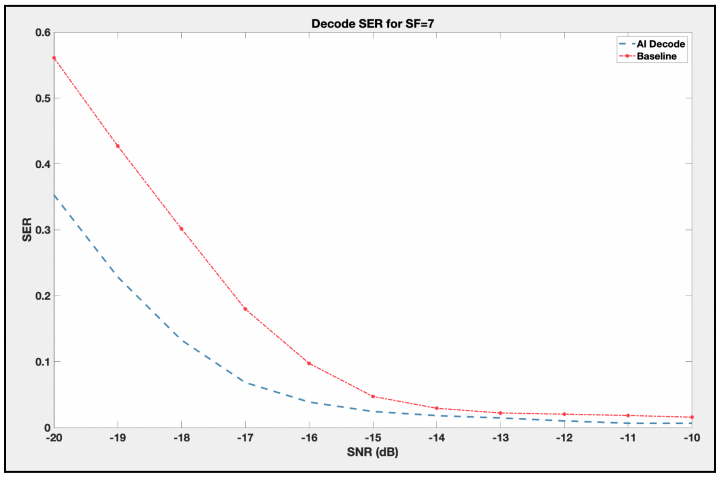
Plot 1.2

- As compared with Plot 1.1, the following plot shows that both the methods AI Decode and Baseline report lower SER values. At SNR (db) of -20 value, the SER value for AI and Baseline is 0.35 and 0.55 respectively as opposed to 0.80 and 1.00.



Plot 1.3

- **Report and submission**
    - Illustrate the design of your model
        - The design of our model as discussed above consists of updated batch size for better training for the Neural Enhanced LoRa model.
        - We were able to release comparable results based on SNR list and training parameters modification since the given model was a baseline neural network.
    - Evaluate your model and compare the performance with baseline. Show the figure and data in your report.
        - Following plot shows the comparison and their respective values.

Plots	Parameter
	For AI Decode, at -20db SNR value, SER = 0.8 For Baseline, at -20db SNR value, SER = 1.00
	For AI Decode, at -20db SNR value, SER = 0.35 For Baseline, at -20db SNR value, SER = 0.55

    - Upload your source code
        - Attached in the zip file.

- **References**
  - [1] https://github.com/hanqingguo/NELoRa-Sensys/tree/main/neural_enhanced_demodulation
  - [2] https://cse.msu.edu/~caozc/papers/sensys21-li.pdf