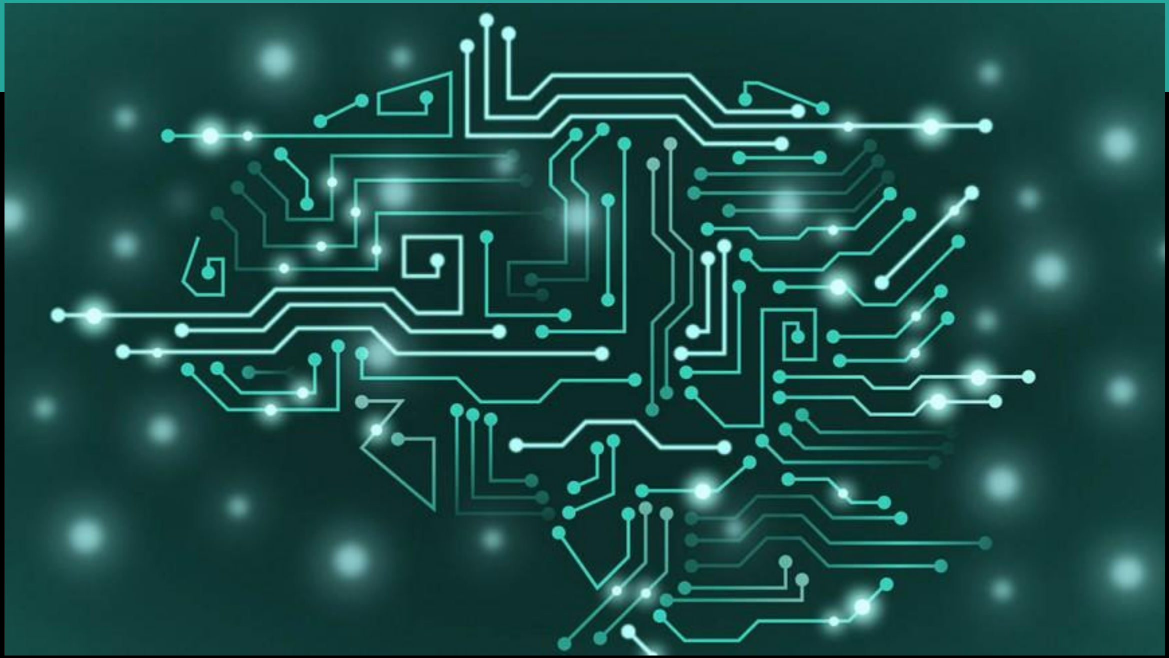
ML Bandits

Deep Learning for Channel Coding via Neural Mutual Information Estimation

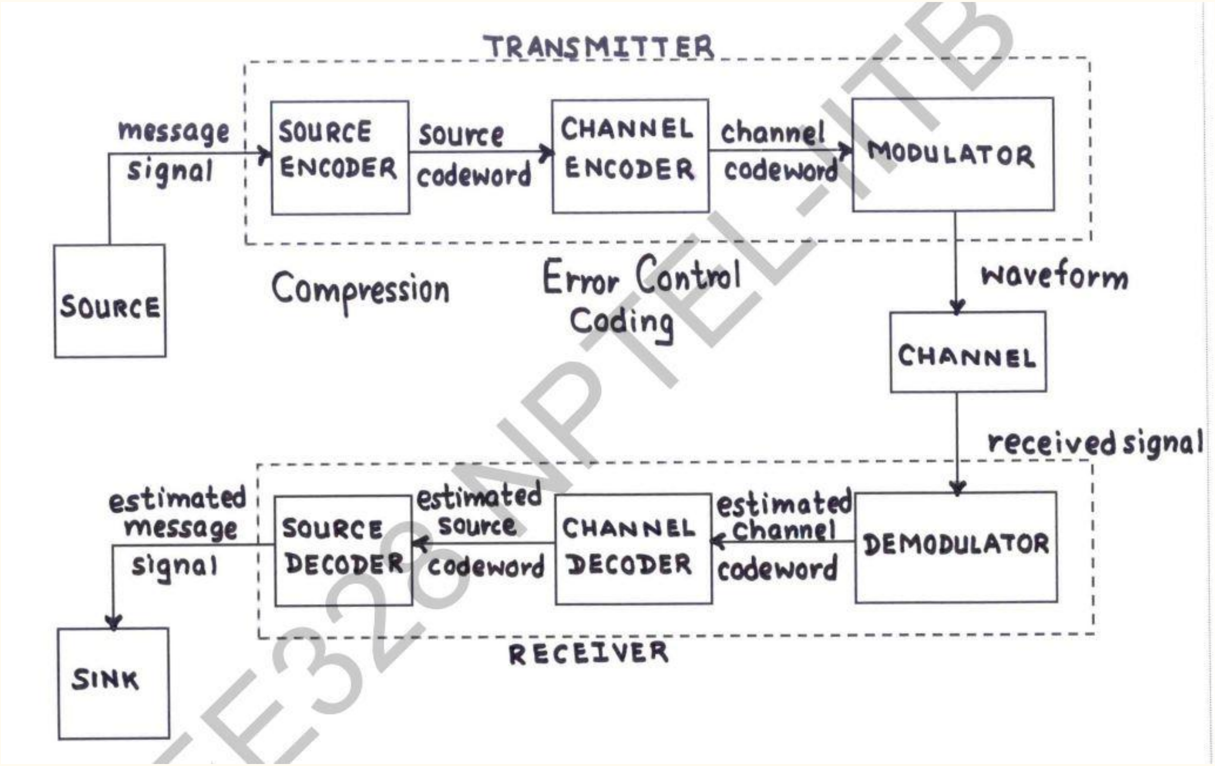


Parth Dodhia
(180070041)

Shaan ul Haque
(180070053)

Course - CS 419(M)
Instructor - Abir De

Generic Communication model



Source : IITB NPTEL course on digital communication

End-to-end deep learning for communication systems

The complete process of training the embeddings for the messages, dense NN to convert the embeddings into signal constellation and training the decoder to decode the received corrupted signal is carried out at one go.

These encoder-decoder systems can achieve bit error rates which come close to practical baseline techniques if they are used for over-the-air transmissions. This is promising since complex encoding and decoding functions can be learned on the fly without extensive communication-theoretic analysis and design, possibly enabling future communication systems to better cope with new and changing channel scenarios.

The basic idea

From a communication theoretic perspective, we know that the optimal transmission rate is a function of the mutual information I(X;Y) between input X (encoding of the original message) and output Y of a channel p(y|x). This optimum transmission rate is called as Capacity of the channel which is maximum I(X;Y) over the input distribution p(x). For additive white Gaussian channel it is given by:

$$\mathbf{C = max_{p(x):\, E[X^{\wedge}2] <= P}\ I(X;Y) = log(1+P/\boldsymbol{\sigma}^2)}$$

This suggests to use mutual information as a metric to learn the optimal channel encoding function of AWGN channels as well as other communication channels.

The Devil and The Messiah

The Mutual Information also depends on the channel's probability distribution, which in most cases is unknown. Thus, instead of approximating the probability distribution of the channel we will approximate the mutual information between input and output of the channel. For this estimation we will employ **Deep Learning**.

Mutual Information - Mathematically

The mutual information is best understood in terms of the KL-divergence (Kullback-Leibler divergence) as given.

$$\begin{aligned} I(X;Y) &:= \int_{\mathcal{X}\times\mathcal{Y}} p(x,y)\log\frac{p(x,y)}{p(x)p(y)}dxdy \\ &= D_{KL}(p(x,y)||p(x)p(y)) \\ &= \mathbb{E}_{p(x,y)}\left[\log\frac{p(x,y)}{p(x)p(y)}\right]. \end{aligned}$$

But we will use other form of it, also known as Donsker-Varadhan representation:

$$D_{KL}(P||Q) = \sup_{g:\Omega\to\mathbb{R}} \mathbb{E}_P[g(X,Y)] - \log(\mathbb{E}_Q[e^{g(X,Y)}])$$

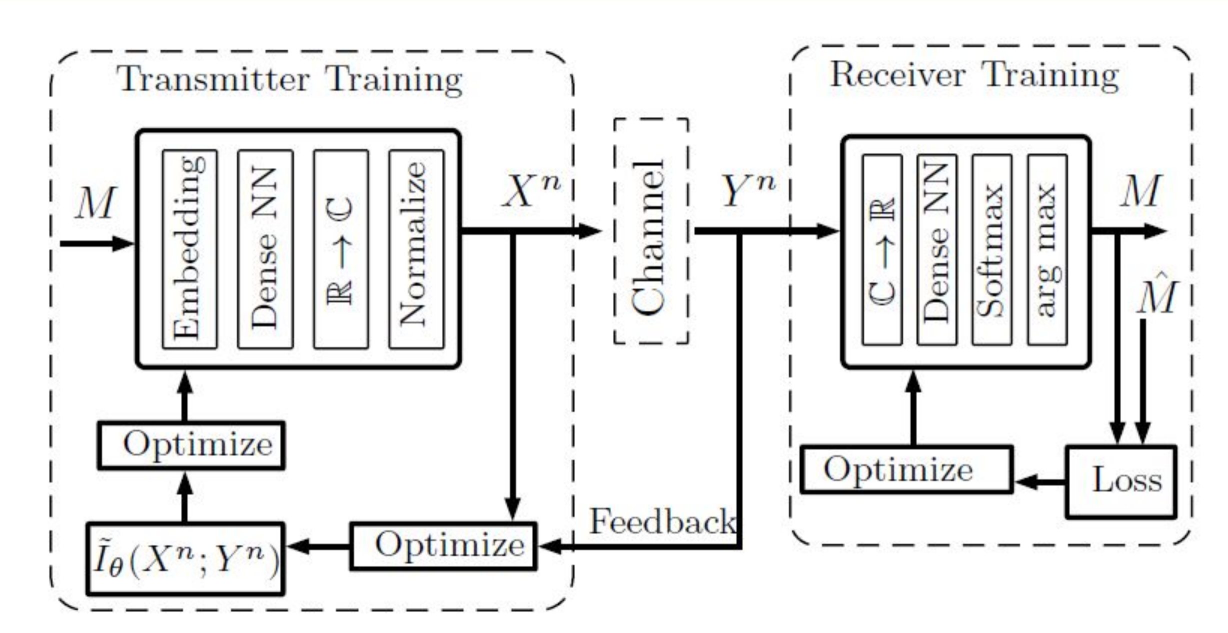
where the supremum is taken over all measurable functions g such that the expectation is finite.

Mutual Information Estimator

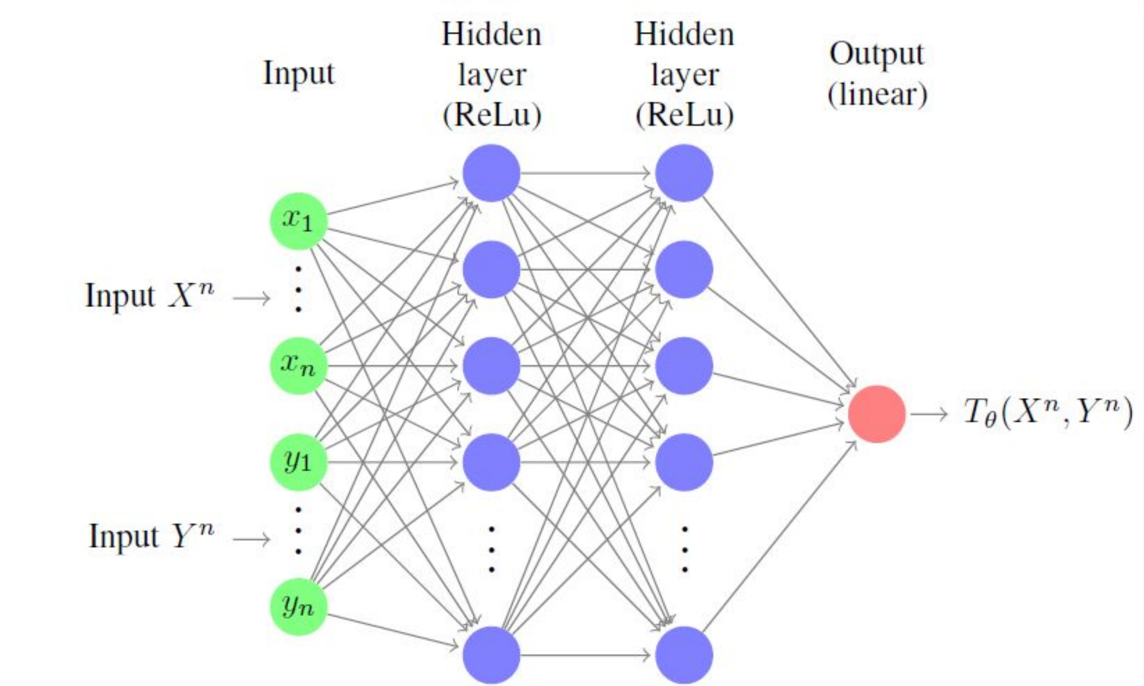
Since Mutual Information is the supremum over all functions, we can choose a neural network parametrized by **θ ϵ Θ** as a family of functions T_{θ}: $X \times Y \rightarrow \mathbf{R}$ for the lower bound. This yields the estimator:

$$I(X;Y) \geq \sup_{\theta \in \Theta} \mathbb{E}_{p(x,y)}[T_\theta(X,Y)] - \log \mathbb{E}_{p(x)p(y)}[e^{T_\theta(X,Y)}]$$

Complete Encoder-Decoder Model



Mutual Information Neural Estimator - MINE



But if distribution is not known then how are we taking expectation??

Let us say k samples were passed through the channel and corresponding k corrupted outputs were observed then we use the empirical mean to estimate the actual mean.

$$\tilde{I}_\theta(X^n;Y^n) := \frac{1}{k}\sum_{i=1}^{k}[T_\theta(x^n_{(i)}, y^n_{(i)})] - \log\frac{1}{k}\sum_{i=1}^{k}[e^{T_\theta(x^n_{(i)}, \bar{y}^n_{(i)})}],$$

First term is mean over joint distribution p(x,y) which is simply passing the k (x_i, y_i) pairs through the neural network. For the second term which mean over marginals, we take the k(k-1) (x_i, y_j) pairs, where $i \neq j$. This is because each signal was generated iid uniformly.

Parameters of the model

Parameters	Encoder	MINE	Decoder
# of layers	4	3	2
Activation	Embedding, Elu, None, Norm Layer	ReLU, ReLU, None	Elu, Softmax
# of nodes	M(=16), M, 2	256, 256, 1	M, M

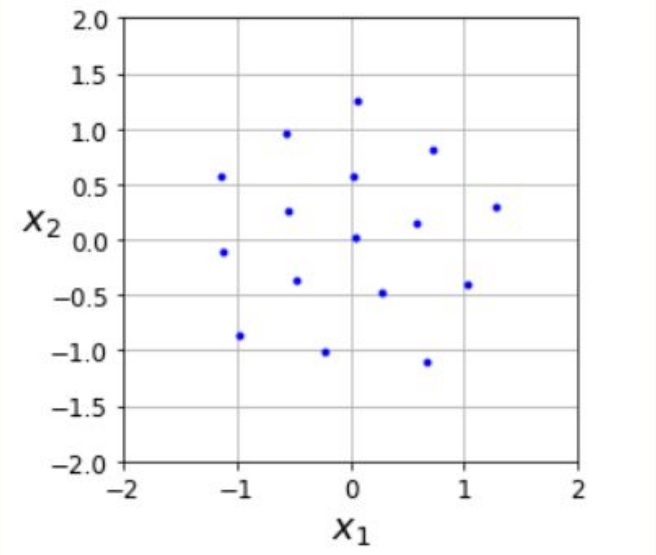
Training Method

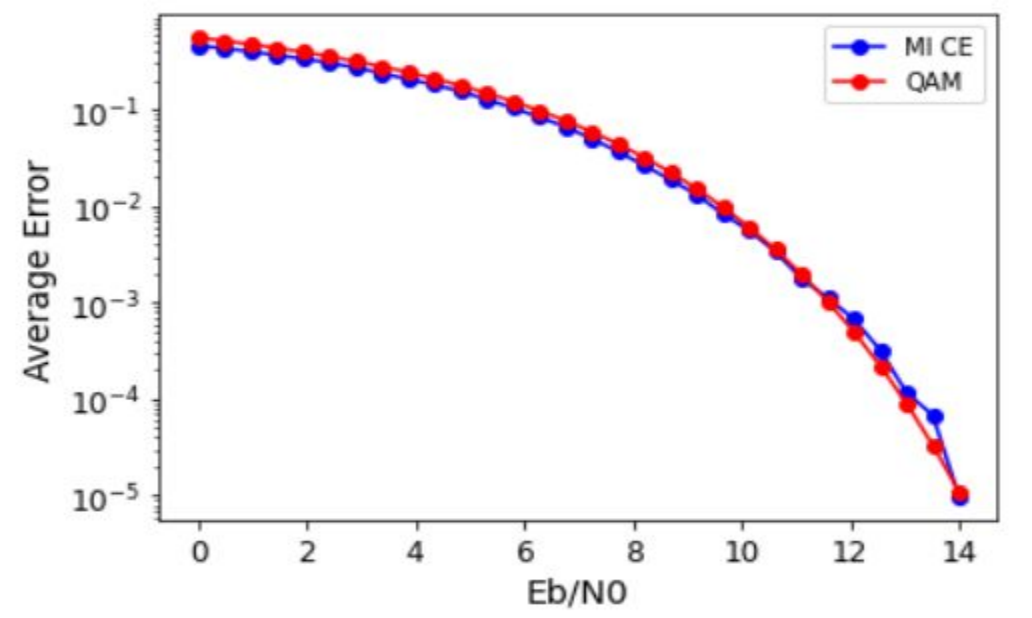
For training SNR was equal to 7dB. We trained encoder and MINE alternately in small batch sizes of 64, with weights updating 20 times in a single epoch. This alternate training went for a total of 10 iterations. The basic goal of our neural network is:

$$\max_{\phi} \max_{\theta} \tilde{I}_{\theta}(X_{\phi}^{n}(m); Y^{n}).$$

After learning the constellations, we trained the decoder. Batch size of 512 was used with with weights updating 500 times in a single epoch.

Learned Constellation and Results





Conclusion

We observe that without going into the complex mathematical analysis or knowing the channel distribution, channel encoding can be done efficiently with error rate similar (rather slightly less for low SNR) to the conventional encoding schemes. Literature review suggests that not only for M=16 or for lower values of M but this scheme works for large number of message signals also with similar results. One of the major drawback of this approach is large number of sample size for accurate mean estimation. This assumption may not be true always and it remains an open question on how to tackle situations where the number of samples are also low.

Images for equations and Model are taken from the paper itself

- R. Fritschek, R. F. Schaefer and G. Wunder, "Deep Learning for Channel Coding via Neural Mutual Information Estimation," 2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), 2019, pp. 1-5, doi: 10.1109/SPAWC.2019.8815464.