Learning to Decode

Kai Niu
niukai@bupt.edu.cn

Key Laboratory of Universal Wireless Communication, Ministry of Education
Beijing University of Posts and Telecommunications (BUPT)

October 11, 2017

Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

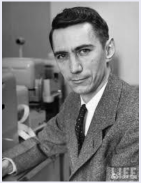
Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

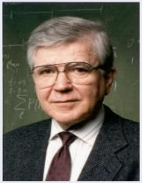
Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

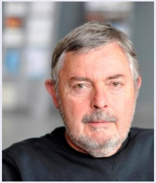
A brief history of channel coding



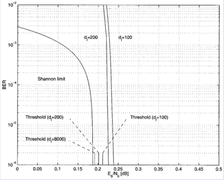
C.E. Shannon
Definition
1948



R.G. Gallager
LDPC Codes
1960



C. Berrou
Turbo Codes
1993
0.7dB



S.Y. Chung
LDPC Codes
2001
0.0045dB

E. Arıkan
Polar Codes
2009
0dB

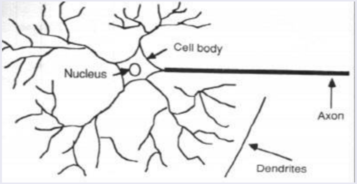
- C. Shannon, "A mathematical theory of communication," *Bell System Tech. J.*, pp. 379-423, July 1948.
- R. G. Gallager, *Low-Density Parity-Check Codes*, Cambridge, MA, MIT Press, 1963.
- C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: turbo-codes," *IEEE Trans. Communications*, pp. 1261-1271, October 1996.
- S.-Y. Chung, J. G. D. Forney, T. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Commun. Lett.*, vol. 5, pp. 58-60, Feb. 2001.
- E. Arıkan, "Channel polarization: A method for constructing capacity achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051-3073, Jul. 2009.

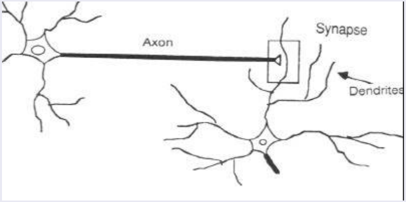
Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Background of Neural Network

The Origin of Neural Network

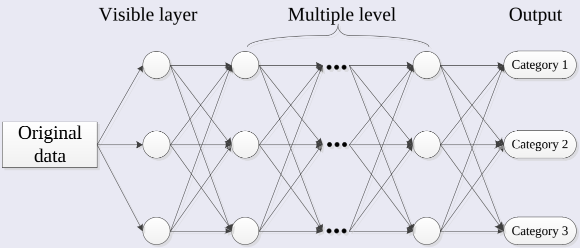




- In the human brain, a typical neuron collects signals from others through a host of fine structures called dendrites.
- The neuron sends out spikes of electrical activity through a long, thin stand known as an axon, which splits into thousands of branches.
- At the end of each branch, a structure called a synapse converts the activity from the axon into electrical effects that inhibit or excite activity in the connected neurons.
- The primary appeal of neural networks is their ability to emulate the brains pattern-recognition skills. The combination of neural network and other techniques, such as distributed storage, parallel processing and nonlinear mapping can get better application results.

Overview of Neural Network

Neural Network Model



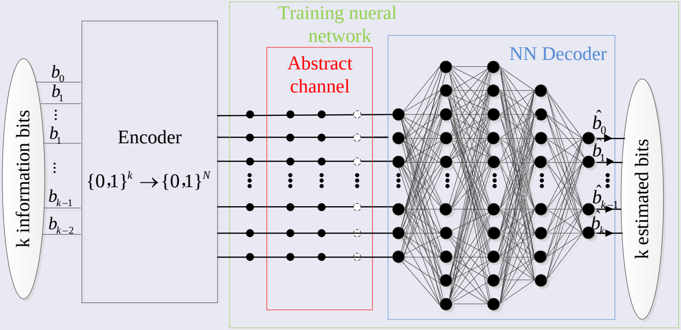
- 1943 the neurophysiologist Warren McCulloch of the University of Illinois and the mathematician Walter Pitts of the University of Chicago proposed the theoretical basis of neural networks.
- In 1954 Belmont Farley and Wesley Clark of the Massachusetts Institute of Technology succeeded in running the first simple neural network.
- Also in 1982, there was a joint US-Japan conference on Cooperative/Competitive Neural Networks. Japan announced a new Fifth Generation effort on neural networks. As a result, there was more funding and thus more research in the field.

Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Neural Network for channel coding

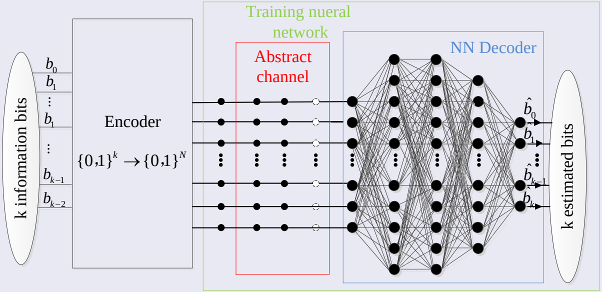
Deep learning setup for channel coding



- At the transmitter, k information bits are encoded into a codeword of length N.
- The coded bits are modulated and transmitted over a noisy channel.
- At the receiver, a noisy version of the codeword is received and the task of the NN decoder is to recover the corresponding information bits base on the experience obtained from training.

Neural Network for channel coding

Learning to code system model



- Learning to decode usually involves two stages of training and decoding.
  - First, the neural network decoder is trained so as to learn the information as much as possible when the input is run through the encoder and then the channel is transmitted.
  - In the decoding phase, the neural network decodes the received signal based on the information and experience obtained during training.

Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

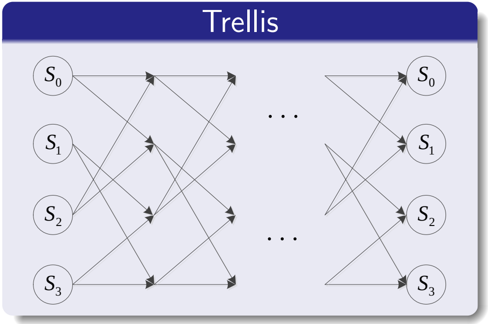
Viterbi Algorithm Overview

History of Viterbi

* Andrew J. Viterbi
* Qualcomm Chief Scientist
* Father of CDMA

- Proposed by Andrew J. Viterbi in 1967
- Maximum likelihood decoding for convolutional codes
- Dynamic programming algorithm,and is generally described by the hidden Markov model (HMM)

Viterbi Algorithm Overview



- All options for each step save the minimum total cost (or maximum value) of all previous steps to the current step, and the previous steps for the current cost;
- After all the steps are calculated, the best choice path is found by back-tracking;
- Conform to this model can be solved by Viterbi Algorithm.

Overview

1 Background
- Channel Coding Overview
- Background of Neural Network
- Learning to Decode

2 Neural Networks for Viterbi Decoder
- Viterbi Algorithm Overview
- Neural Network for Viterbi Algorithm
- Simulation

3 Neural Networks for Belief Propagation
- Belief Propagation Overview
- Application Belief Propagation on LDPC
- Neural Network on Belief Propagation

4 Neural Networks Decoder for Polar Coding
- Polar Coding Principle
- Deep Learning-based Decoding of Polar Codes
- Simulation

5 Summary

6 References

Neural Network for Viterbi Algorithm

Two basic designs

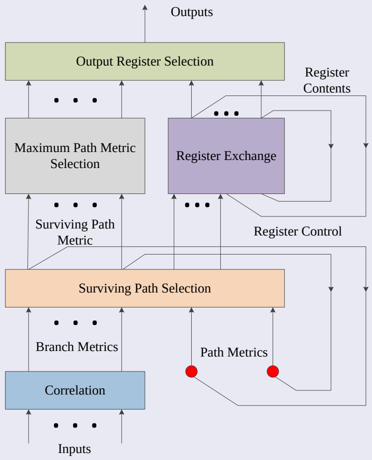
- Register Exchange
  - * A shift register is associated with every state in the decoding trellis
  - * The register for a given state at a given time contains the information bits associated with the surviving partial path that terminates at that state
  - * The registers are updated and exchanged as dictated by the surviving branches
- Trace Back
  - * To search forward according to the preferred paths
  - * When exploring a certain state, finding that previous choice is not the best or fail to reach the target, it is a step back to re-select

Neural Network for Viterbi Algorithm

Decoder modules:

* Input correlation
* Surviving path selection
* Register exchange
* Maximum path metric selection
* Output register selection

The block diagram of ANN decoder

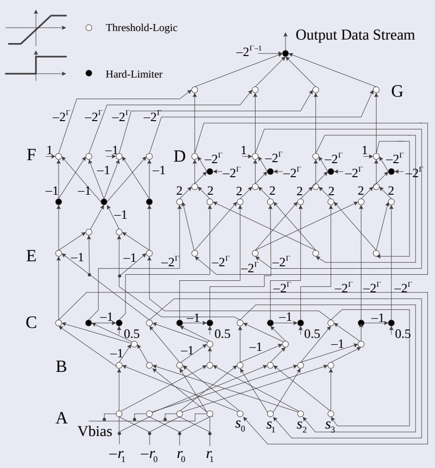


Neural Network for Viterbi Algorithm

- Modules can be implemented using threshold-logic and hard-limiter neurons

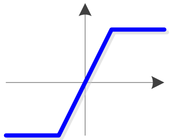
* Input correlation: Level A
* Surviving path selection: Level B and C
* Register exchange: Level D
* Maximum path metric selection: Level E and F
* Output register selection: Level G

Rate $1/2$, $K = 3$

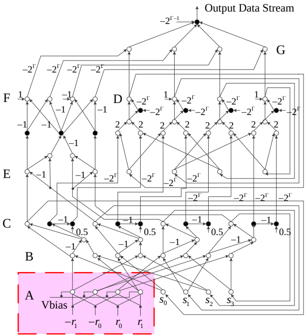


Level A

- Compute the inner products of received signals
- A bias constant is added to ensure the positive branch metric, since TL neurons have a zero output when the input is negative



Threshold-Logic



Neural Network for Viterbi Algorithm

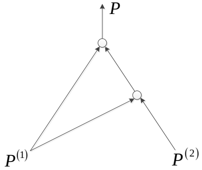
Level B

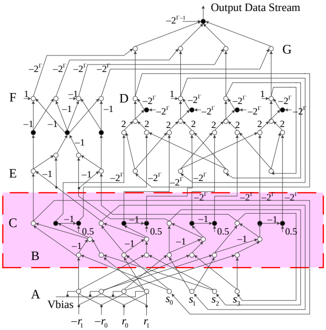
- Add the branch metrics computed by level A to the partial path metrics
- Generate new path metrics for the next sates

$$P^{(1)} = b_1 + P_{i_1}$$

$$P^{(2)} = b_2 + P_{i_2}$$

$$\begin{aligned} P_i &= max\{P^{(1)}, P^{(2)}\} \\ &= P^{(1)} + max\{P^{(2)} - P^{(1)}, 0\} \end{aligned}$$





Neural Network for Viterbi Algorithm

Level C

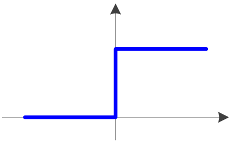
- Send the surviving paths to the register-exchange module
- Each pair of HL neurons is associated with two paths entering a given trellis state
    - ∗ With the surviving path:output$= 0$, input to the register exchange module$= 0$
    - ∗ Others:output$= 1$, input to the register exchange module$= -2^{\Gamma}$

these values control the register exchange

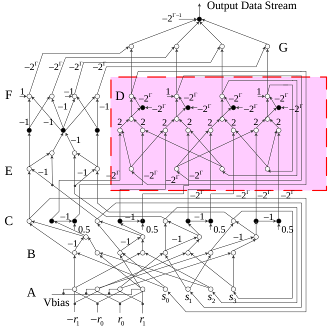
Neural Network for Viterbi Algorithm

Level D

- Store Γ bits of information in form of an integer in the range $[0, 2^{\Gamma} - 1]$
- These values are changed as dictated by the HL neurons from the surviving paths



Hard-Limiter



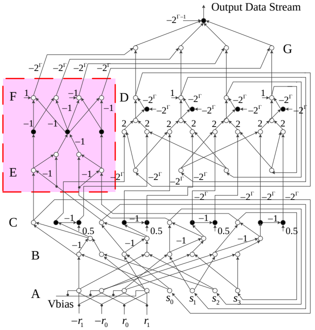
Neural Network for Viterbi Algorithm

Level E

- Two neurons in the left collect the current path metrics for the four states
- Determine the maximum value

Level F

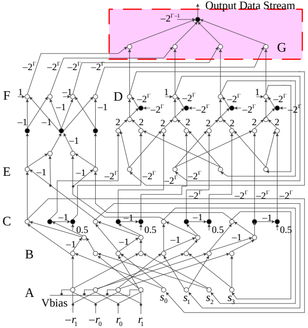
- Decoding the HL neurons
- Output with the maximum path metric is 0,else is 1



Neural Network for Viterbi Algorithm

Level G

- Select one of four neuron outputs and add its integer to $-2^{(\Gamma-1)}$
- The most significant bit of the selected register determine the ANN decoder output

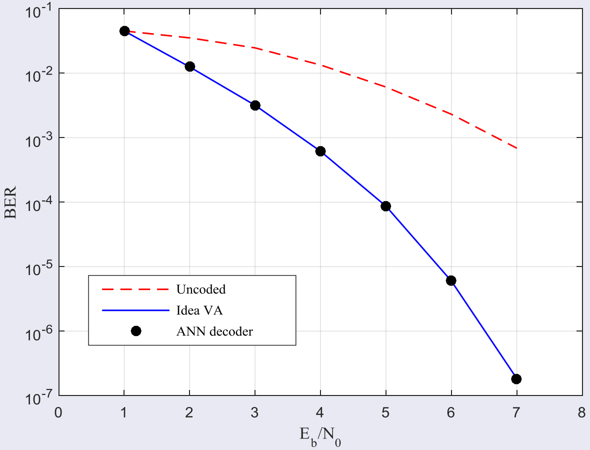


Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Simulation

AWGN channel,rate $1/2$,number of neurons= 514[1][W.Xiaoan 1996]



- The performance of ANN decoder exactly matches that of Ideal VA.

Conclusion

Advantages

- Using analog, artificial neurons, and the structure does not place any limit on the speed.
- The TL and HL neurons are easily implemented using existing linear and nonlinear components
- The complexity of ANN decoder can be fully determined by the parameters of convolutional codes

Distribution of neurons in ANN Viterbi decoder

	TL neuron	HL neuron
Input correlation	2^n	
Path metric feedback	2^M	
Survival path select	$2^{M+k}+2^M 2\left(2^k-1\right)$	$2^M\left(2^k-1\right)$
MINNET	$2\left(2^M-1\right)$	

k:input bit n:output bit M:number of memory elements

Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Belief Propagation Overview

Robert G. Gallager

- American electrical engineer.
- a member of the technical staff at the Bell Telephone Laboratories.
- known for his work on information theory and communications networks.
- received the Claude E. Shannon Award from the IEEE Information Theory Society.

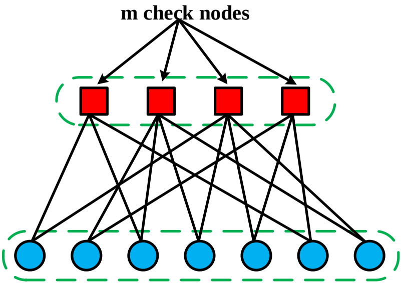
Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

LDPC Encoder Introduction

- The LDPC code defined by the graph is the set vectors $\mathbf{C}{=}(c_1, \ldots c_n)$ such that $\mathbf{H}c^T = 0$
- example:

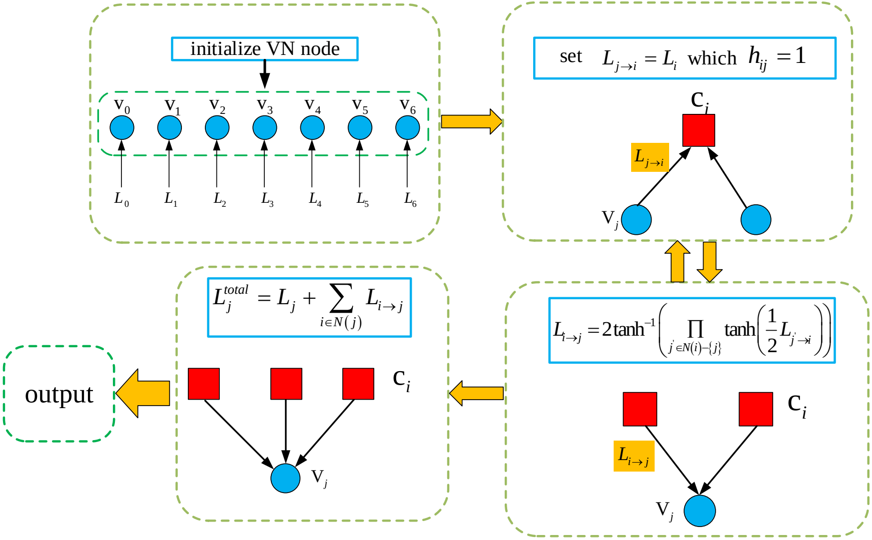
$$\mathbf{Hc}^T = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$



Belief Propagation

- The Sum-Product algorithm is usually called Belief Propagation when the messages represent beliefs.
- For the BEC, BP boils down to the following very simple rules.
  - At variable nodes, if all of the messages used for the computation of the outgoing message are erasures, then the outgoing message is an erasure.
  - At check nodes, if at least one of the messages used for the computation of the outgoing message is an erasure, then the outgoing message is an erasure.

Application Belief Propagation on LDPC



BP Algorithm for LDPC Decode

Algorithm 1 The Gallager Belief Propagation Algorithm

Input: initialize L_j and set $L_{j\to i}{=}L_j$ for which $h_i j{=}1$;

1: **CN update:** Compute outgoing CN messages $L_{i\to j}$ for each CN using:

$$L_{i\to j} = 2\tanh^{-1}\left(\prod_{j'\in N(i)-\{j\}} \tanh\left(\tfrac{1}{2}L_{j'\to i}\right)\right)$$

2: **VN update:** Compute outgoing VN messages $L_j \to i$ for each VN using Equation.

$$L_{j\to i} = L_j + \sum_{i\prime\in N(j)-\{i\}} L_{i\prime\to j}$$

3: **LLR total:**

4: **for** $j = 0 : n-1$ **do**

5: $\quad L_j^{total} = L_j + \sum_{i\in N(j)} L_{i\to j}$

6: $\quad \hat{v}_j = \begin{cases} 1, & \text{if} L_j^{total} < 0 \\ 0, & \text{else} \end{cases}$

7: **end for**

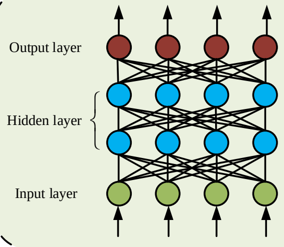
8: if $\hat{v}\mathbf{H} = 0$ or the number of iterations equals the maximum limit,stop else go to Step 2.

9: **return** v.

Overview

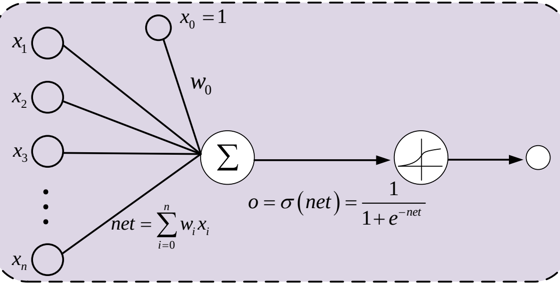
1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Neural Network Architectures

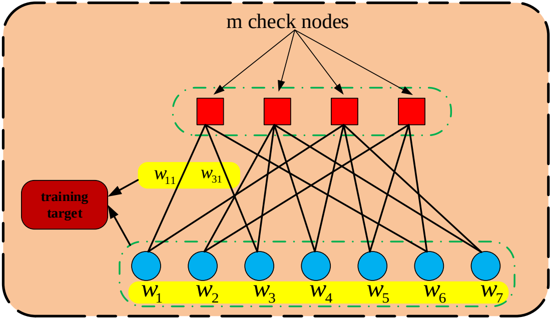


$$y_i = \tanh\left(\sum_j W_{ij} x_j + W_{iO}\right)$$

$$y_i = \sigma\left(\sum_j W_{ij} x_j + W_{iO}\right) = \frac{1}{1+\exp\left[-\left(\sum_j W_{ij} x_j + W_{iO}\right)\right]}$$

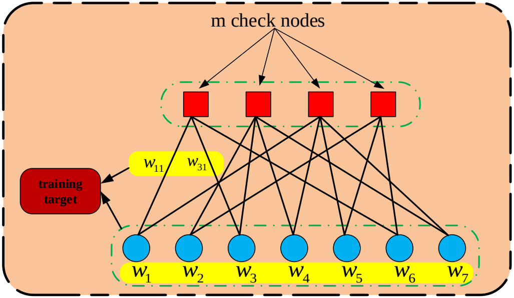


Application Deep Learning to Decode



- use the trellis representation for the decoder as in the BP algorithm. The difference is that assigning weights to the edges in the Tanner graph.
- Hence by optimal setting (training) of the parameters of the neural network, its performance can not be inferior to plain BP.

Application Deep Learning to Decode

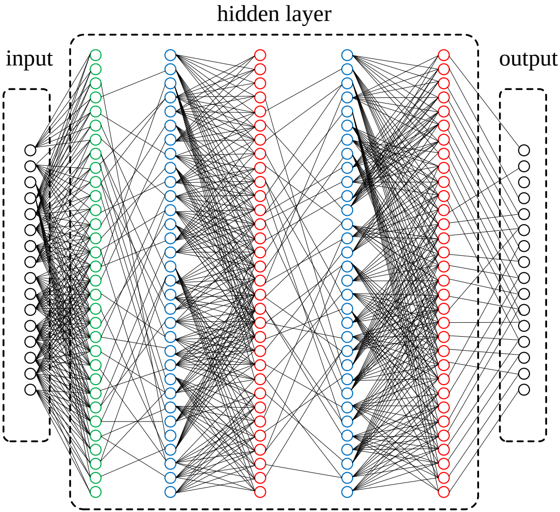


The goal is to train the parameters $\{w_i, w_{i,j}\}$ to achieve an N dimensional output word.

- These weights will be trained using stochastic gradient descent which is the standard method for training neural networks.

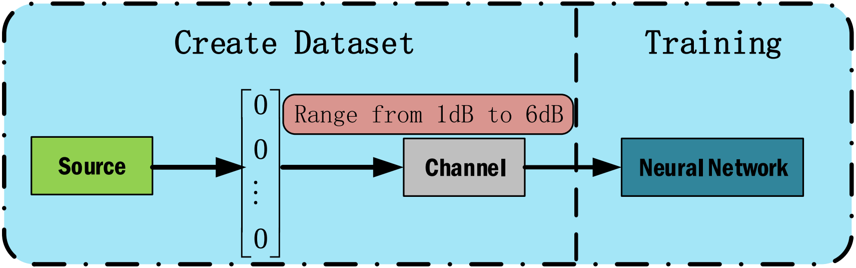
$$L_{i\to j} = 2\tanh^{-1}\left(\prod_{j'\in N(i)-\{j\}} \tanh\left(\frac{1}{2}\left(w_i L_i + \sum_{i'\in N(j)-\{i\}} w_{i,j} L_{j'\to i}\right)\right)\right)$$

Application Deep Learning to Decode



BCH(15,11) with 5 hidden layers which correspond to 3 full BP iterations.

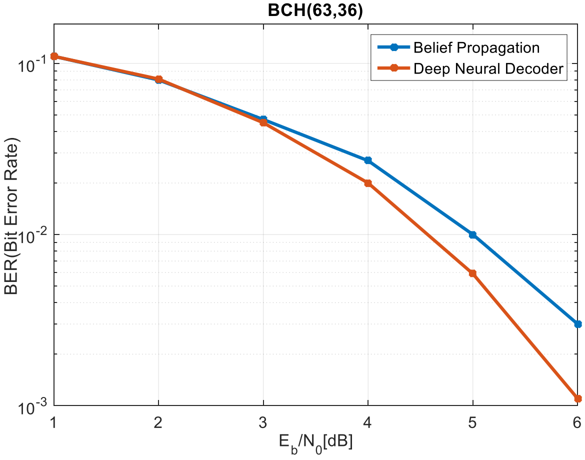
Deep Learning Training Data



Training parameters:

- the training data are all zero codeword.
- SNRs ranging from 1dB to 6dB.
- each mini batch has 20 codewords for each SNR.

Deep Learning Training Data[2][E. Nachmani 2016]



Simulation Parameters:

- channel type: AWGN.
- type is BCH(63,36).
- SNRs ranging from 1dB

Conclusions

From the simulation results and analysis,it's obvious that:

- improve performance compared to plain BP without increasing the required computational complexity.
  eg:achieve the same BER performance of 50 iteration BP with 5 iteration of the deep neural decoder, This is equal to complexity reduction of factor 10.
- it's suitable for short BCH codes, for larger BCH codes, the BP algorithm and the deep neural network still have a significant gap from maximum likelihood.

Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Overview of Polar Coding

Polar encoding

- Block code.
- Both non-systematic and systematic encoding can be implemented.

Construction

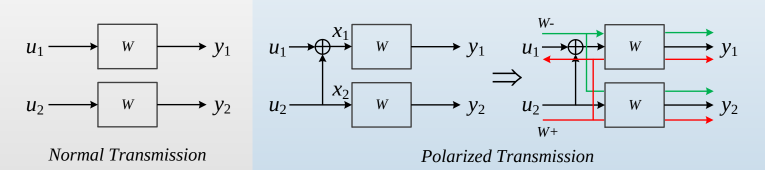
- BEC channel: Bhattacharyya parameters calculation.
- Density evolution (DE) and Tal & Vardy algorithm.
- Gaussian approximation.

Decoding

- Successive cancellation (SC) decoding: low complexity.
- Improved SC decoding: high performance.

The Basic Idea of Channel Polarization

Transforms N independent copies of a given channel W into a second set of channels that show a polarization effect.

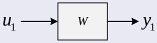


- Two copies of a binary input channel $W : \mathbb{F}_2 \to \mathcal{Y}$
- Consider the transformation above to generate two channels $W^- : \mathbb{F}_2 \to \mathcal{Y}^2$ and $W^+ : \mathbb{F}_2 \to \mathcal{Y}^2 \times \mathbb{F}_2$

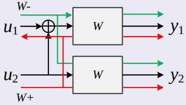
$$I\left(W^-\right) \leq I\left(W\right) \leq I\left(W^+\right) \tag{1}$$

$$I\left(W^-\right) + I\left(W^+\right) = 2I\left(W\right) \tag{2}$$

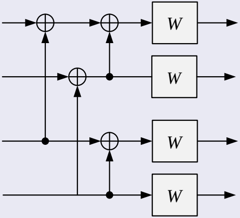
Channel polarization transform



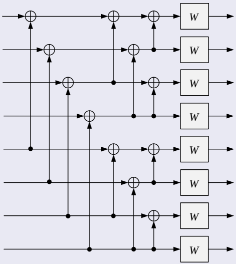
One channel transmission



Two channel polarization



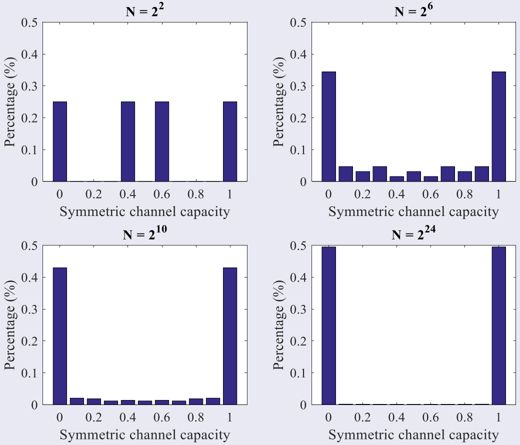
Four channel polarization



Eight channel polarization

Channel Polarization Transform

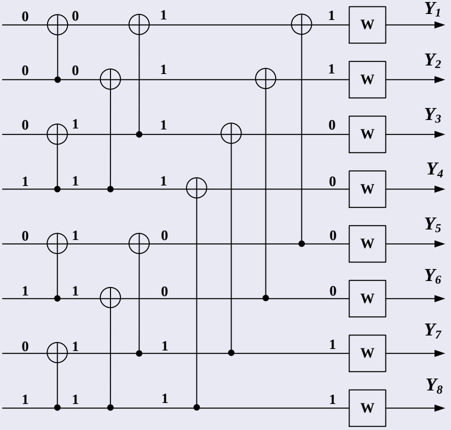
Polarization phenomenon demonstration



- $*$ BEC channel with erasure probability 0.5;
- $*$ The capacities are shown on the X-axis;
- $*$ The number of subchannels in each capacity interval is represented by the height.

$(8, 4)$ polar code for BEC channel with $\varepsilon = 0.5$

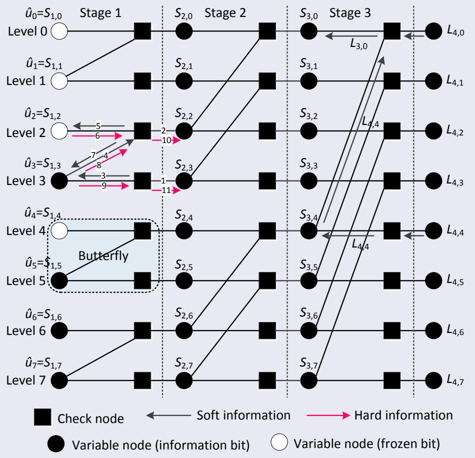
$I(W_i)$	Rank		
0.0039	8	*frozen*	0
0.1211	7	*frozen*	0
0.1914	6	*frozen*	0
0.6836	4	*Data*	U_4
0.3164	5	*frozen*	0
0.8086	3	*Data*	U_6
0.8789	2	*Data*	U_7
0.9961	1	*Data*	U_8



- Encoding complexity: $O(N \log N)$.

Successive Cancellation (SC) Decoding

Trellis representation of polar codes



Soft Messages Updated Rule:

* In check nodes, $\left\lfloor \frac{j-1}{2^{i-1}} \right\rfloor \bmod 2 = 0$, and

$$L_{i,j} = f\left(\frac{L_{i+1,j}}{2}, \frac{L_{i+1,j+2^{i-1}}}{2}\right) \qquad (3)$$

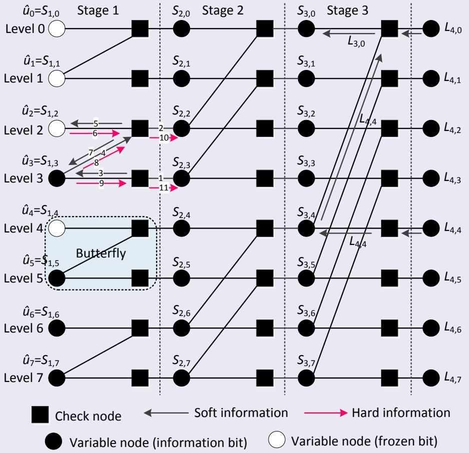
where $f(x,y) = 2\tanh^{-1}\left[\tanh(x)\cdot\tanh(y)\right]$.

* In variable nodes, $\left\lfloor \frac{j-1}{2^{i-1}} \right\rfloor \bmod 2 = 1$, and

$$L_{i,j} = \left(1 - 2s_{i,j-2^{i-1}}\right) L_{i+1,j-2^{i-1}} + L_{i+1,j} \qquad (4)$$

Successive Cancellation (SC) Decoding

Trellis representation of polar codes



Hard Messages Updated Rule:

- When $\left\lfloor \frac{j-1}{2^{i-1}} \right\rfloor \bmod 2 = 0$,

$$S_{i+1,j} = S_{i,j} \oplus S_{i,j+2^{i-1}} \qquad (5)$$

where $\oplus$ is the modulo-2 operation.

- When $\left\lfloor \frac{j-1}{2^{i-1}} \right\rfloor \bmod 2 = 1$,

$$S_{i+1,j} = S_{i,j} \qquad (6)$$

Complexity:

- Require $\mathcal{O}\left(N \log_2 N\right)$ processing elements.

Decoding of Polar Codes

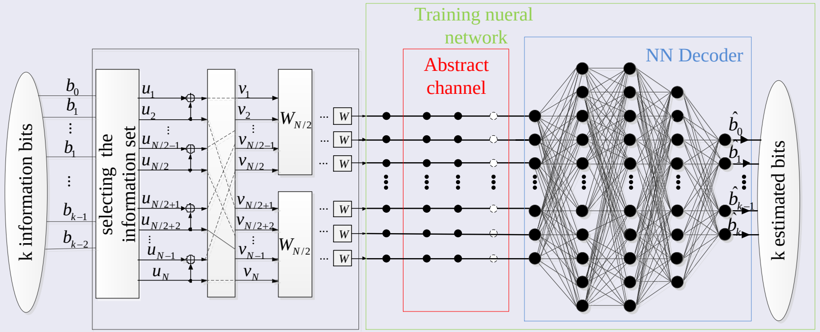
- Successive cancellation (SC) decoding: A deep-first search method with complexity roughly $\mathcal{O}\left(N \log N\right)$.
  - Sufficient to prove that polar codes achieve capacity.
  - Equivalent to an earlier algorithm by Schnabl and Bossert (1995) for RM codes.
- List decoding: A breadth-first search algorithm with limiting branch.
  - First proposed by Tal and Vardy (2011) for polar codes [17, 18].
  - List decoding was used earlier by Dumer and Shabunov (2006) for RM codes.
  - Complexity grows as $\mathcal{O}\left(LN \log N\right)$ for a list size L.
- Sphere-decoding ("British Museum" search with branch and bound, starts decoding from the opposite side) [24][K. Niu].
- CRC aided list decoding: CRC helps to select the correct decoding path from the list, which makes polar codes perform better than LTE turbo codes with the comparable complexity [20, 21, 22][K. Niu].

Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Basic Deep Learning for Polar coding

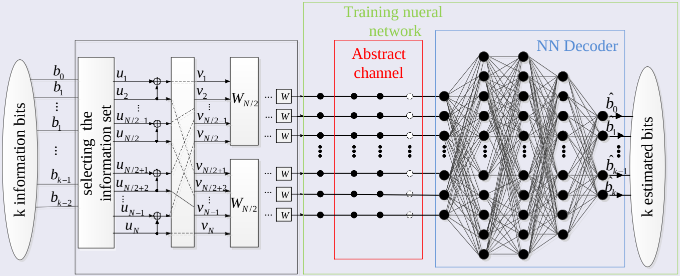
Deep learning setup for channel coding



- A NN consists of many connected neurons. In such a neuron all of its weighted inputs are added up, a bias is optionally added, and the result is propagated through a nonlinear activation function, e.g., a sigmoid function or a rectified linear unit (ReLU), which are respectively defined as:

$$g_{sigmoid}(\mathcal{Z}) = \frac{1}{1+e^{-\mathcal{Z}}}, g_{\text{relu}}(\mathcal{Z}) = \max\{0, \mathcal{Z}\} \tag{7}$$

Deep learning setup for channel coding



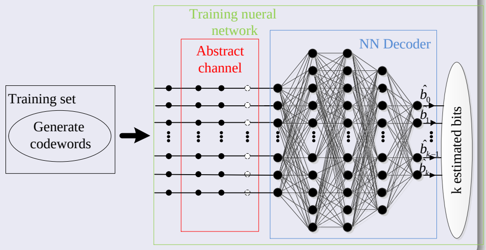
- Denoting v as input and w as output of the NN, an input-output mapping is defined by a chain of functions depending on the set of parameters θ by

$$w = f(v;\theta) = f^{(L-1)}(f^{(L-2)}(\cdots(f^{(0)}(v)))) \tag{8}$$

where L gives the number of layers and is also called *depth*.

Deep Learning for Polar coding

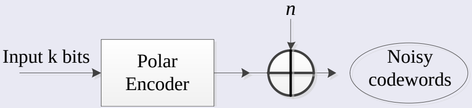
Nueral network training of polar codes



- A training set of known input-output mappings is required and a specific loss function has to be defined so as to find the weights of NN.
- Weights of the NN can be found to minimize the loss function over the training set,by the use of gradient descent optimization methods and the backpropagation algorithm
- The goal of the training is to enable NN to find the correct output for the unknown input.

Deep Learning for Polar decoding

Generate training data for polar



- As is show in the above fig, k information bits are encoded into a codeword of length N. Then the coded bits are transmitted through the noise channel, and the original code bits are added with noise.
- the desired NN output is easy to obtained when noisy codewords are generated and the transmitted information bits are obviously known.
- There are as many known input-output mapping as NN required corresponding each k information bits for the noisy is easy to generate for free.

Deep Learning for Polar decoding

Polar code training process

- It is possible to extend the NN with additional layers for modulating and adding noise so as to keep the training set small.
- Sigmoid function(7)forces the output neurons to be in between zero and one, which can be interpreted as the probability that a "1" was transmitted.
- A loss function is a function that maps an event onto a real number intuitively representing some "cost" associated with the event. Examples for such loss functions:

$$\mathrm{L}_{\mathrm{MSE}}=\frac{1}{k}\sum_i \left(b_i-\hat{b}_i\right)^2 \tag{9}$$

$$\mathrm{L}_{\mathrm{BCE}}=-\frac{1}{k}\sum_i \left[b_i \ln(\hat{b}_i)+(1-b_i)\ln(1-\hat{b}_i)\right] \tag{10}$$

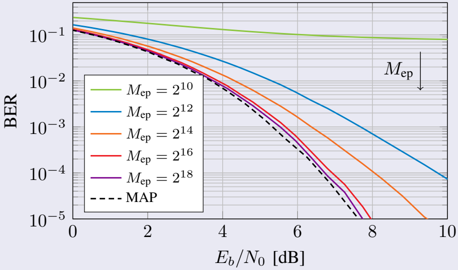
where $b_i \in \{0,1\}$is the ith target information bit (label) and $\hat{b}_i \in [0,1]$ the NN soft estimate.

Overview

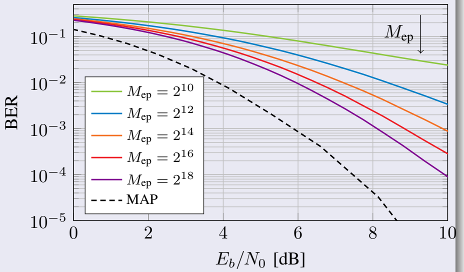
1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Influence of the number of epochs M_{ep} on the BER

Polar Code[3][T. Gruber 2017]



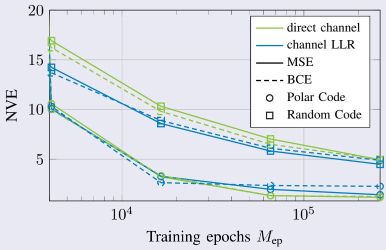
Random Code[3][T. Gruber 2017]



- NN decoder employing three hidden layers with 128, 64, and 32 nodes.
- For both code families, the larger the number of training epochs, the closer is the gap between MAP and NND performance.
- For polar codes, close to MAP performance is achieved for $M_{ep} = 2^{18}$ epochs, while we may need a larger NN or more training epochs for random codes.

Comprehensive Comparison of Learning to Decode

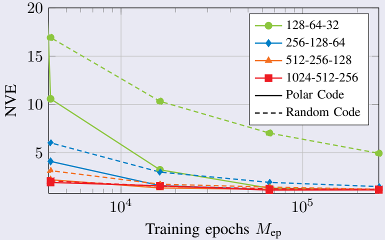
Learning curve for 16 bit-length codes with code rate r = 0:5 for a 128-64-32 NN.[3][T. Gruber 2017]



- The influence of direct channel values versus channel LLR values as decoder input in combination with two loss functions, MSE and BCE.
  - For a few training epochs, the LLR input improves the learning process.
  - This advantage disappears for a larger M_{ep}.The same holds for BCE against MSE.
  - For polar codes with LLR values and BCE the learning appears not to converge for the applied number of epochs.
- In summary, for training the NN with a large number of training epochs it does not matter if LLR or channel values are used as inputs and which loss function is employed. Moreover, normalization is not required.

Comprehensive Comparison of Learning to Decode

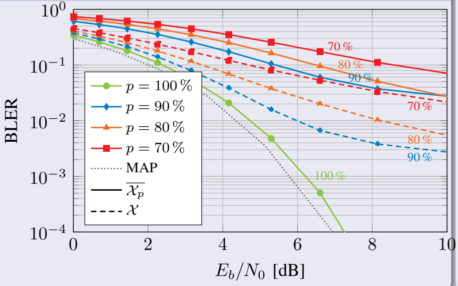
Learning curve for different NN sizes for 16 bit-length codes with code rate r = 0:5.[3][T. Gruber 2017]



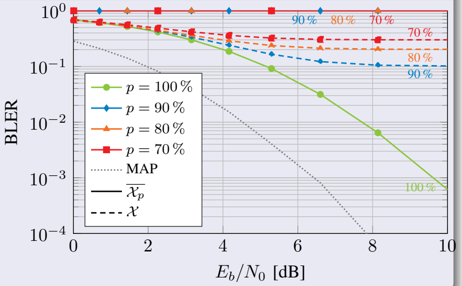
- For both polar and random codes, it is possible to achieve MAP performance.
- The larger the net, the less training epochs are necessary.
- In general, the larger the number of layers and neurons, the larger is the expressive power or capacity of the NN

Influence of the number of epochs M_{ep} on the BER

Polar Code[3][T. Gruber 2017]



Random Code[3][T. Gruber 2017]



- BLER for a 128-64-32 NN trained on Xp with $\chi_p = 2^{18}$ learning epochs[3][T. Gruber 2017] . Solid and dashed lines show the performance on $\tilde{\chi}_p$ on χ, respectively.
- While for polar codes the NN is able to decode codewords that were not seen during training, the NN cannot decode any unseen codeword for random codes.

General Remarks

- For small block lengths, neural network achieved to decode random codes as well as polar codes with MAP performance.
- Learning to decode is limited through exponential complexity as the number of information bits in the codewords increases.
- The neural network is able to generalize to codewords that it has never seen during training for structured, but not for random codes.
- The neural network is able to generalize for structured codes, which means decoding algorithms can be learned.

Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

Summary

Theoretical contribution

- Neural network decoder inherently describes a highly parallelizable structure, enabling one-shot decoding. This renders deep learning-based decoding a promising alternative channel decoding approach as it avoids sequential algorithms.
- Nueral network may generalize to structured codewords that it has never seen during training.

Learning to decode

- The Neural network is able to decode the channel codes base on a small training set.
- The neural network decoder can be generalized from a small training set to the whole codeword net.
- The neural network decoder can be generalized between different type channel codes.

Summary

Disadvantages of learning to decode

- Neural network depends on structured codewords. For random code, it shows an awful performance.
- Neural network needs to process training with a lot of training sample, which lead to extra overload.
- The training complexity of deep learning-based channel decoders scales exponentially with the codebook size and therefore with the number of information bits.
- Learning to decode is currently only feasible for very short block lengths.

Overview

1. Background
   - Channel Coding Overview
   - Background of Neural Network
   - Learning to Decode
2. Neural Networks for Viterbi Decoder
   - Viterbi Algorithm Overview
   - Neural Network for Viterbi Algorithm
   - Simulation
3. Neural Networks for Belief Propagation
   - Belief Propagation Overview
   - Application Belief Propagation on LDPC
   - Neural Network on Belief Propagation
4. Neural Networks Decoder for Polar Coding
   - Polar Coding Principle
   - Deep Learning-based Decoding of Polar Codes
   - Simulation
5. Summary
6. References

References I

[1] [W.Xiaoan 1996]
Xiao-an Wang and Stephen B. Wicker, "An Artificial Neural Net Viterbi Decoder," *IEEE Trans. Com.*, vol. 44, no. 2, pp. 165-171, Feb. 1996.

[2] [E. Nachmani 2016]
E. Nachmani, Y. Be'ery and D. Burshtein, "Learning to decode linear codes using deep learning," *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 341-346, Sept. 2016.

[3] [T. Gruber 2017]
T. Gruber, S. Cammerer, J. Hoydis and S. t. Brink, "On deep learning-based channel decoding," *2017 51st Annual Conference on Information Sciences and Systems (CISS), Baltimore*, pp. 1-6, May. 2017.

[4] [A. Hamalainen 1999]
A. Hamalainen and J. Henriksson, "A recurrent neural decoder for convolutional codes," *IEEE International Conference on Communications*, vol. 2, No. 99CH36311, pp. 1305-1309, June. 1999.

[5] [Yu-Jhih Wu 1995]
Yu-Jhih Wu, P. M. Chau and R. Hecht-Nielsen, "A supervised learning neural network coprocessor for soft-decision maximum-likelihood decoding," *IEEE Transactions on Neural Networks*, vol. 6, no. 4, pp. 986-992, Jul 1995.

[6] [A. R. Karami 2009]
A. R. Karami, M. Ahmadian Attari and H. Tavakoli, "Multi Layer Perceptron Neural Networks Decoder for LDPC Codes," *2009 5th International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1-4, Sept. 2009.

[7] [Xiao-An Wang 1996]
Xiao-An Wang and S. B. Wicker, "An artificial neural net Viterbi decoder," *IEEE Transactions on Communications*, vol. 44, no. 2, pp. 165-171, Feb. 1996.

[8] [E. Arıkan 2009]
E. Arıkan, "Channel polarization: A method for constructing capacity achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051-3073, Jul. 2009.

[9] [B. Li 2015]
B. Li, H. Shen and D. Tse, "A RM-Polar Codes," *arXiv:1407.5483 [cs.IT]*, Jul. 2014.

[10] [E. Arıkan 2011]
E. Arıkan, "Systematic Polar Coding," *IEEE Communications Lett.*, vol. 15, no. 8, pp. 860-862, Aug. 2011.

[11] [E. Arıkan 2008]
E. Arıkan, "A performance comparison of polar codes and Reed-Muller codes," *IEEE Commun. Lett.*, vol.12, no.6, pp.477-479, Jun. 2008.

References III

[12] [R. Mori 2009]
R. Mori and T. Tanaka, "Performance of polar codes with the construction using density evolution," *IEEE Commun. Lett.*, vol. 13, no. 7, pp. 519-521, Jul. 2009.

[13] [I. Tal and A. Vardy 2013]
I. Tal and A. Vardy, "How to Construct Polar Codes," *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp.6562-6582, Oct. 2013.

[14] [P. Trifonov 2012]
P. Trifonov, "Efficient Design and Decoding of Polar Codes," *IEEE Trans. Commun.*, vol.60, no.11, pp.3221-3227, Nov. 2012.

[15] [S.-Y. Chung 2012]
S.-Y. Chung, T. J. Richardson and R. L. Urbanke, "Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation," *IEEE Trans. Inf. Theory*, vol.47, no.2, pp.657-670, Feb. 2001.

[16] [J. Dai and K. Niu 2016]
J. Dai, K. Niu, Z. Si and J. Lin, "Evaluation and Optimization of Gaussian Approximation for Polar Codes," *arXiv:1511.07236v2 [cs.IT], submitted to IEEE Transactions on Communications*, May 2016.

References IV

[17] [I. Tal and A. Vardy 2011]
I. Tal and A. Vardy, "List Decoding of Polar Codes," in *Proc. 2011 IEEE International Symposium on Information Theory (ISIT)*, pp. 1-5, 2011.

[18] [I. Tal and A. Vardy 2015]
I. Tal and A. Vardy, "List decoding of polar codes," *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213-2226, May. 2015.

[19] [Z. Zhang 2015]
Z. Zhang, L. Zhang, X. Wang, C. Zhong and H. V. Poor, "A Split-Reduced Successive Cancellation List Decoder for Polar Codes," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 2, pp. 292-302, Feb. 2016.

[20] [K. Niu 2012]
K. Niu and K. Chen, "CRC-Aided Decoding of Polar Codes," *IEEE Commun. Lett.*, vol. 16, no. 10, pp. 1668-1671, Oct. 2012.

[21] [K. Chen and K. Niu 2013]
K. Chen, K. Niu and J. Lin, "Improved Successive Cancellation Decoding of Polar Codes," *IEEE Transactions on Communications*, vol. 61, no. 8, pp. 3100-3107, Aug. 2013.

References V

[22] [K. Niu 2014]
K. Niu, K. Chen, J. Lin and Q.T. Zhang, "Polar codes: Primary concepts and practical decoding algorithms," *Communications Magazine, IEEE*, vol. 52, no. 7, pp. 192-203, Feb. 2014.

[23] [B. Li 2014]
B. Li, H. Shen, and D. Tse, "An adaptive successive cancellation list decoder for polar codes with cyclic redundancy check," *IEEE Commun. Lett.*, vol. 16, no. 12, pp. 2044-2047, 2012.

[24] [K. Niu 2014]
K. Niu, K. Chen and J. Lin, "Low-Complexity Sphere Decoding of Polar Codes Based on Optimum Path Metric," *IEEE Commun. Lett.*, vol.18, no.2, pp.332-335, Jul. 2014.

[25] [K. Niu 2013]
K. Niu, K. Chen, and J. Lin, "Beyond turbo codes: rate-compatible punctured polar codes," in *Proc. 2013 IEEE Int. Conf. on Communications (ICC)*, pp. 2016-2020, 2013.

[26] [K. Chen and K. Niu 2013]
K. Chen, K. Niu and J. Lin, "A hybrid ARQ scheme based on polar codes," *IEEE Communications Lett.*, vol.17, no.10, pp. 1996-1999, Oct. 2013.

The End, Thanks!