Markov Chain Neural Networks

Maren Awiszus · Bodo Rosenhahn

Gottfried Wilhelm Leibniz Universität Hannover

Introduction

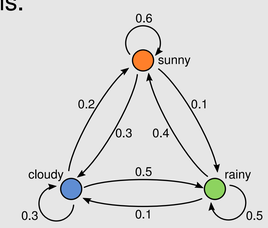
Neural Networks:

- Approximates any function
- Non-linear combination of weights through activations
- Deterministic

Markov Chains:

- Models stochastic process
- Discrete state space with transition probabilities
- Non-deterministic

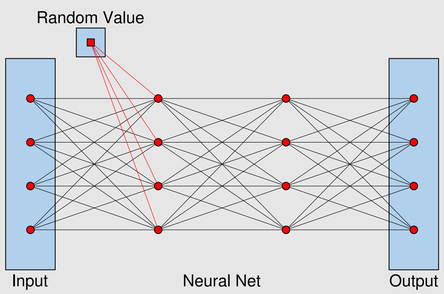
Our Goal: Allow non-deterministic behavior for Neural Networks in order to model Markov Chains.



Markov Chain Example

Method

Idea: Add an additional input variable which contains a random value. This random value will act as a switch to ensure different outcomes.

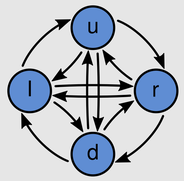


For that, the training data must be distributed to ranges for this value during training.

Results

Random Walker

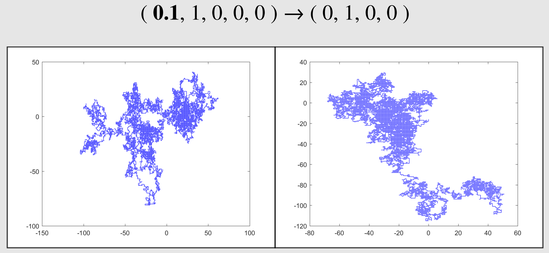
A typical example for a Markov Chain is a Random Walker:



$$\begin{pmatrix} 0 & 1/3 & 1/3 & 1/3 \\ 1/3 & 0 & 1/3 & 1/3 \\ 1/3 & 1/3 & 0 & 1/3 \\ 1/3 & 1/3 & 1/3 & 0 \end{pmatrix}$$

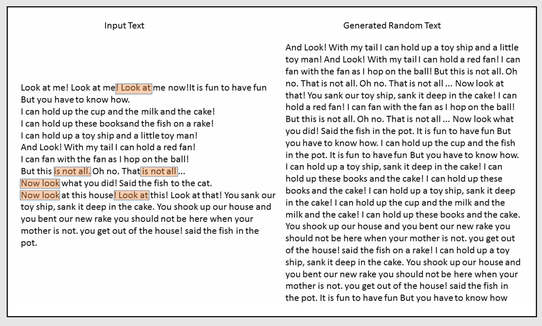
Training with random value:

$$(\ \mathbf{0.5}, 1, 0, 0, 0\ ) \rightarrow (\ 0, 0, 1, 0\ )$$
$$(\ \mathbf{0.2}, 1, 0, 0, 0\ ) \rightarrow (\ 0, 1, 0, 0\ )$$
$$(\ \mathbf{0.8}, 1, 0, 0, 0\ ) \rightarrow (\ 0, 0, 0, 1\ )$$
$$(\ \mathbf{0.9}, 1, 0, 0, 0\ ) \rightarrow (\ 0, 0, 0, 1\ )$$
$$(\ \mathbf{0.1}, 1, 0, 0, 0\ ) \rightarrow (\ 0, 1, 0, 0\ )$$



Text Synthesis

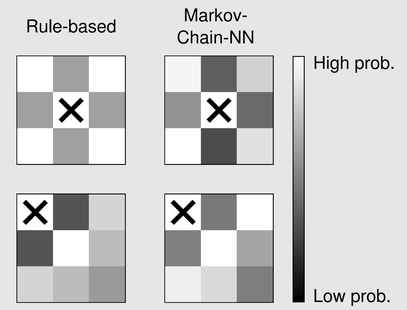
MCNN can generate different continuations for given repetitions. Here an example from Dr. Seuss' "The Cat in the Hat":



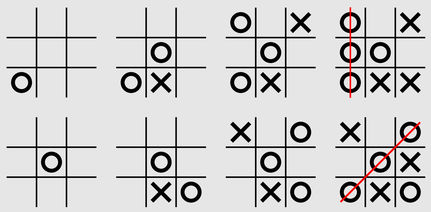
Input Text	Generated Random Text
Look at me! Look at me! Look at me nowIt is fun to have fun But you have to know how. I can hold up the cup and the milk and the cake! I can hold up these booksand the fish on a rake! I can hold up a toy ship and a little toy man! And Look! With my tail I can hold a red fan! I can fan with the fan as I hop on the ball! But this is not all. Oh no. That is not all... Now look at what you did! Said the fish to the cat. Now look at this house! Look at this! Look at that! You sank our toy ship, sank it deep in the cake. You shook up our house and you bent our new rake you should not be here when your mother is not. you get out of the house! said the fish in the pot.	And Look! With my tail I can hold up a toy ship and a little toy man! And Look! With my tail I can hold a red fan! I can fan with the fan as I hop on the ball! But this is not all. Oh no. That is not all... Now look at that! You sank our toy ship, sank it deep in the cake! I can hold a red fan! I can fan with the fan as I hop on the ball! But this is not all. Oh no. That is not all ... Now look what you did! Said the fish in the pot. It is fun to have fun But you have to know how. I can hold up the cup and the fish in the pot. It is fun to have fun But you have to know how. I can hold up a toy ship, sank it deep in the cake! I can hold up these books and the cake! I can hold up these books and the cake! I can hold up a toy ship, sank it deep in the cake! I can hold up the cup and the milk and the milk and the cake! I can hold up these books and the cake. You shook up our house and you bent our new rake you should not be here when your mother is not. you get out of the house! said the fish on a rake! I can hold up a toy ship, sank it deep in the cake. You shook up our house and you bent our new rake you should not be here when your mother is not. you get out of the house! said the fish in the pot. It is fun to have fun But you have to know how

Tic-Tac-Toe

With a non-deterministic Neural Net, the same step in a game can have different outcomes:

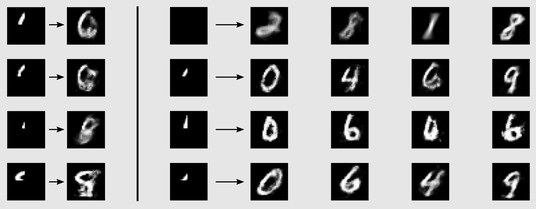


Two different outcomes with MCNN as circle:



MNIST data completion

While a standard NN will generate a mixture of possible data, MCNN will generate different possible solutions.



Standard NN MCNN

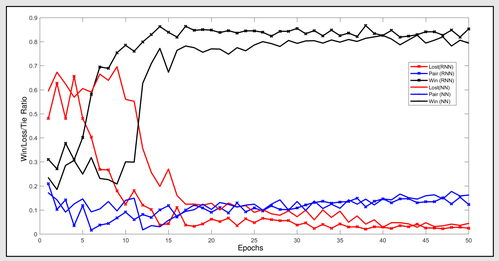
Summary and Conclusions

We present a modified Neural Network model which is capable to simulate Markov Chains. We show how to train such a network and demonstrate applications:

- Standard MC models
- Non-deterministic behavior in games
- Data completion and -synthesis

The MCNN:

- uses a random variable as a switch node to produce different outcomes
- is based on a statistical analysis of the training data
- does not require further post-processing (e.g. sampling from distrib.)
- is straight-forward to implement
- converges faster than a comparable NN (results from Tic-Tac-Toe experiment):



Interested in our other projects?
Visit us at:
www.tnt.uni-hannover.de

Institut für Informationsverarbeitung
Maren Awiszus
awiszus@tnt.uni-hannover.de
http://www.tnt.uni-hannover.de/∼awiszus

Vision with Biased or Scarce Data (VBSD) 2018
in conjunction with CVPR 2018, Salt Lake City

Leibniz Universität Hannover