Toward Biologically Inspired Deep Learning: Facilitation-Depression Recurrent Neural Networks

Maceo Richards

Abstract:

Recurrent neural networks (RNNs) are a class of deep learning algorithms designed to handle sequence data and are widely used across many domains of research. They work by processing sequence data one element at a time while maintaining a hidden state. This hidden state allows the algorithm to retain information from previous computational steps regarding context and relationships within the data. While RNNs proved to be a great innovation for modeling sequence data, their internal operations are difficult to comprehend. This paper presents a novel RNN algorithm inspired by presynaptic short-term facilitation and depression dynamics of spiking biological neurons. A first advantage of this is that the algorithm internal operations are set within a biological context, which enables their intuitive comprehension. A second advantage may be realized by analyzing how a framework describing presynaptic short-term facilitation and depression creates diversity in node types and their respective expressions. This work is motivated by an effort to emulate naturally emergent intelligence in artificial learning algorithms.

Background:

Liquid state machines (LSMs) are a class of recurrent algorithms outside of deep learning, initially proposed by Maass and Markham (2004). These algorithms consist of a collection of nodes which artificially replicate dynamics of cortical microcircuits having diverse types of neurons and synaptic connections. Each node receives time varying stimulation from both the algorithm inputs and other nodes, and a spatio-temporal pattern of activations within the network of nodes emerges. Although Maass and Markham (2004) pose LSMs to be an improvement over deep learning algorithms due to their diversity of node types, they have some crucial drawbacks. LSMs operate in real-time, and significant amounts of time may be required for a meaningful pattern of activations to emerge from the network nodes. An ideal recurrent algorithm would retain the LSM advantage of node type diversity while eliminating this drawback.

A framework better describing short-term temporal patterns must be employed to overcome this drawback. This framework must also describe a biological process with diverse components to retain advantages offered by LSMs. Lee, Anton, Poon and McRae (2009) theorized such a framework proposing a unified model for presynaptic short-term facilitation and depression dynamics of spiking neurons. In the context of synaptic plasticity, facilitation and depression refer to how the strength of synapses increase and decrease, respectively, under repeated stimulation. These phenomena are largely attributed to presynaptic processes such as calcium-dependent release of neurotransmitters and depletion of presynaptic vesicles. Lee et al. (2009) use the definitions in Table 1 and equations 1-7 to explain how their unified model of presynaptic short-term facilitation and depression determine the elicitation of excitatory postsynaptic current (EPSC) from spiking neurons.

The unified model put forth by Lee et al. (2009) is comprised of differential equations and constitutes a dynamical system. This means that a RNN algorithm inspired by this model will not only be adept at learning temporal patterns within sequence data but also require differential equation solvers in its internal operations. A work in support of this is that of liquid time-constant (LTC) networks, a type of RNN algorithm developed by Hasani, Lechner, Amini, Rus and Grosu (2021). Instead of using implicit nonlinearities to determine dynamics of a learning system, LTC networks learn first-order derivative

Table 1 Nomenclature of symbols used by Lee et al. (2009) to describe presynaptic short-term facilitation and depression dynamics of spiking neurons.

Symbol	Unit	Definition
I_{Ca}	Hz	injected calcium current
$\delta(\cdot)$	-	Dirac delta function
N_p	#	number of stimulus impulses
τ_{Ca_i}	ms	calcium decay time constant
Ca_i	μM	intracellular calcium concentration
Ca_{i0}	μM	intracellular calcium concentration at rest
K_{Ca}	$\mu M s^{-1}$	intracellular gain in calcium concentration per action potential
P_{rel}	-	probability of vesicle release
$P_{rel,max}$	-	maximum probability of vesicle release
$K_{rel,1/2}$	uM	calcium concentration at half occupancy
k_{recov}	ms^{-1}	recovery rate from empty to releasable state
k_{recov0}	ms^{-1}	minimum recovery rate from empty to releasable state
$k_{recov,max}$	ms^{-1}	maximum recovery rate from empty to releasable state
$K_{recov,1/2}$	μM	calcium concentration yielding one half recovery rate
R_{rel}	-	remaining ratio of vesicles releasable
$Flux_{Glu}$	# ms^{-1}	flux of glutamate release
Glu	#	number of neurotransmitter glutamate molecules
n	#	number of glutamate molecules released per vesicle
N_{total}	#	total number of vesicles per synaptic terminal
τ_{EPSC}	ms	EPSC decay time constant
K_{Glu}	$\mu M s^{-1}$	gain in glutamate concentration yielded by influx of glutamate

$$I_{Ca} = \sum_{k=1}^{N_p} \delta(t - t_k) \tag{1}$$

$$\tau_{Ca_i} * \frac{dCa_i(t)}{dt} = (-Ca_i + Ca_{i0}) + K_{Ca} * I_{Ca} \tag{2}$$

$$P_{rel} = P_{rel,max} * \frac{{Ca_i}^4}{{Ca_i}^4 + K_{rel,1/2}^4} \tag{3}$$

$$k_{recov} = k_{recov0} + \left(k_{recov,max} - k_{recov0}\right) * \frac{Ca_i}{Ca_i + K_{recov,1/2}} \tag{4}$$

$$\frac{dR_{rel}(t)}{dt} = k_{recov} * (1 - R_{rel}) - P_{rel} * I_{Ca} * R_{rel} \tag{5}$$

$$Flux_{Glu} = \frac{dGlu(t)}{dt} = n * N_{total} * R_{rel} * P_{rel} * I_{Ca} \tag{6}$$

$$\tau_{EPSC} * \frac{dEPSC(t)}{dt} = -EPSC - K_{Glu} * Flux_{Glu} \tag{7}$$

characteristics of dynamical systems regulated by nonlinear interlinking gates and their outputs are computed with numerical differential equation solvers (Hasani et al. 2021). Further, they argue that LTC networks demonstrate improved expressive power over other algorithms in the family of neural ordinary differential equations and result in better performance on applications within the time-series domain (Hasani et al.).

Methods:

To emulate this system of equations as an artificial RNN algorithm, a few small changes are made. The expressive power of this dynamical system should be retained while also minimizing both the number of learnable parameters and required computational steps. Some of the above equations have been reformulated in equations 8-11 to achieve this. For the remainder of this paper, some of the parameter nomenclature suggested by Lee et al. (2009) is changed as well.

Ca_i and τ_{Ca_i} will now be referred to as Ca and τ_{Ca} for simplicity, and Ca_{i0} will be referred to as Ca_{mu} for more intuitive comprehension of the sigmoid (σ) function components in equations 9 and 10. Equation 8 is a reformulation of equation 2 describing the current intracellular calcium concentration Ca, where $\alpha = \frac{K_{Ca}}{\tau_{Ca}}$ and is a learnable parameter for this dynamical system.

$$\tau_{Ca} * \frac{dCa(t)}{dt} = (-Ca + Ca_{mu}) + K_{Ca} * I_{Ca} \rightarrow \frac{dCa(t)}{dt} = \alpha * I_{Ca} - \frac{Ca - Ca_{mu}}{\tau_{Ca}} \tag{8}$$

A slight change to equation 3 characterizing the probability of vesicle release P_{rel} is suggested in equation 9 where the fractional component in the original equation is substituted with a sigmoid function component. A new parameter Ca_{sigma} is proposed and quantifies the significance of how much the current intracellular calcium concentration Ca deviates from the intracellular calcium concentration at rest Ca_{mu}.

$$P_{rel} = P_{rel,max} * \frac{Ca^4}{Ca^4 + K_{rel,1/2}^4} \rightarrow P_{rel} = P_{rel,max} * \sigma\left(\frac{Ca - Ca_{mu}}{Ca_{sigma}}\right) \tag{9}$$

The same substitution is made in equation 10 for the fractional component in equation 4 describing k_{recov}. The parameter k_{recov0} is now referred to as $k_{recov,min}$, as this is its direct interpretation, and the difference $k_{recov,max} - k_{recov,min}$ is quantified by a parameter $k_{recov,\Delta}$.

$$k_{recov} = k_{recov,min} + \left(k_{recov,max} - k_{recov,min}\right) * \frac{Ca}{Ca + K_{recov,1/2}} \rightarrow \tag{10}$$

$$k_{recov} = k_{recov,min} + k_{recov,\Delta} * \sigma\left(\frac{Ca - Ca_{mu}}{Ca_{sigma}}\right)$$

Equation 11, which had previously been summarized in equations 6 and 7, is the full formulation of the differential equation for EPSC where $\beta = \frac{n * N_{total} * K_{Glu}}{\tau_{EPSC}}$ and is also a learnable parameter.

$$\tau_{EPSC} * \frac{dEPSC(t)}{dt} = -EPSC - K_{Glu} * Flux_{Glu} \rightarrow \tag{11}$$

$$\frac{dEPSC(t)}{dt} = -\frac{EPSC}{\tau_{EPSC}} - \beta * R_{rel} * P_{rel} * I_{Ca}$$

It is necessary here to understand the logic behind the substitutions made for the fractional components of equations for P_{rel} and k_{recov}. These equations share two key qualities in the original work presented by Lee et al. (2009), the first being that they each attempt to model natural nonlinear dynamics. The second may be realized by considering what happens for P_{rel} and k_{recov} as the intracellular calcium concentration varies in the original equations. Larger and smaller intracellular calcium concentrations result in respectively larger and smaller vesicle release probabilities. Similarly, vesicle recovery rates increase and decrease as the intracellular calcium concentrations grow and shrink, respectively. These same behaviors are observed in the reformulated equations for P_{rel} and k_{recov}. These alterations to the original equations not only reduce the number of learnable parameters for this dynamical system but also promote expressive power through the inclusion of nonlinearities standard in machine learning.

Discussion:

A recurrent algorithm of this design offers key benefits, first that its internal operations are interpretable from a biological perspective. Lee et al. (2009) state that the resonance frequency of a biological neuron (which is a function of many biophysical parameters and has a closed-form solution under their unified model) is indicative of its inclination for synaptic facilitation or depression under repeated stimulation. Because of this, the behavior of artificial nodes in the recurrent algorithm presented here may be investigated after training and interpreted as tending toward either facilitation or depression of node activations.

A second benefit becomes apparent in realizing how a framework describing presynaptic short-term facilitation and depression defines a spectrum of behaviors that neurons may exhibit in response to a stimulus. The parameters regulating facilitation and depression dynamics allow for a broad variety of activation behaviors and therefore promote diversity in how external information is processed by individual nodes within both natural and artificial learning systems.

Another benefit is that this recurrent algorithm will also have intrinsic cell memory. Many RNNs struggle to handle long-term dependencies in sequential data. Long Short-Term Memory (LSTM) networks are a type of RNN employing a sophisticated architecture with memory cells that allow it to effectively handle long-term dependencies. This cell memory is jointly described by the states of intracellular calcium concentration Ca and remaining ratio of vesicles releasable R_{rel} in the recurrent algorithm presented here. The temporal information contained within these cell states will aid the algorithm in producing a hidden state that denotes EPSC from each artificial neuron. In the context of presynaptic short-term facilitation and depression, initial hidden and cell states before calcium current injection have biological definitions. A function $f(\cdot)$ which converts an input to the recurrent algorithm of dimensionality M into injected calcium current for each of N artificial neurons may be learned.

Emulating naturally emergent intelligence in artificial learning algorithms has many benefits, primarily that it can result in better performing algorithms. Additionally, researchers will be able to explore tenets of biological intelligence from within a computational setting. See the Appendix for a full description of the facilitation-depression recurrent algorithm and view the code implementation at https://github.com/maceor22/Deep-Direct-Discriminative-Decoder.

Appendix:

Definitions:

$*$ = element-wise multiplication

$\cdot$ = matrix multiplication

$dt^{(1\times 1)} = 1/$ode_solver_steps

$x(1{:}T)^{(T\times M)}$ = input datum with dimensionality M and sequence length T

$f\left(x(t)^{(1\times M)}\right)^{(1\times N)} = \exp(x(t)^{(1\times M)} \cdot \gamma^{(M\times N)})$

$I_{Ca}^{(1\times N)}$= injected calcium current

$Ca^{(1\times N)}$ = intracellular calcium concentration

$Ca_{mu}^{(1\times N)}$ = intracellular calcium concentration at rest

$Ca_{sigma}^{(1\times N)}$ = significance of intracellular calcium concentration deviance from rest state

$\tau_{Ca}^{(1\times N)}$ = calcium decay time constant

$\alpha^{(1\times N)}$ = parameter quantifying K_{Ca}/τ_{Ca}

$R_{rel}^{(1\times N)}$ = remaining ratio of vesicles releasable

$EPSC^{(1\times N)}$ = excitatory postsynaptic current

$\tau_{EPSC}^{(1\times N)}$ = EPSC decay time constant

$\beta^{(1\times N)}$ = parameter quantifying $(n * N_{total} * K_{Glu})/\tau_{EPSC}$

$P_{rel}^{(1\times N)}$ = probability of vesicle release

$P_{rel,max}^{(1\times N)}$ = maximum probability of vesicle release

$k_{recov}^{(1\times N)}$ = recovery rate from empty to releasable state

$k_{recov,min}^{(1\times N)}$ = minimum recovery rate from empty to releasable state

$k_{recov,\Delta}^{(1\times N)}$ = parameter quantifying $k_{recov,max} - k_{recov,min}$

Facilitation-Depression Recurrent Algorithm:

```
$Ca = Ca_{mu}, R_{rel} =$ ones vector, $EPSC =$ zeros vector
For i = 1…T do
    $I_{Ca} = f\left(x(i)\right)$
    For j = 1…ode_solver_steps do
        $P_{rel} = P_{rel,max} * \sigma\left(\frac{Ca - Ca_{mu}}{Ca_{sigma}}\right)$
        $EPSC = EPSC - dt * [EPSC/\tau_{EPSC} + \beta * R_{rel} * P_{rel} * I_{Ca}]$
        $k_{recov} = k_{recov,min} + k_{recov,\Delta} * \sigma\left(\frac{Ca - Ca_{mu}}{Ca_{sigma}}\right)$
        $R_{rel} = R_{rel} + dt * [k_{recov} * (1 - R_{rel}) - P_{rel} * I_{Ca} * R_{rel}]$
        $Ca = Ca + dt * [\alpha * I_{Ca} - (Ca - Ca_{mu})/\tau_{Ca}]$
```

References:

Maass, W., & Markram, H. (2004). On the computational power of circuits of spiking neurons. *Journal of computer and systems sciences, 69*(4), 593-616.

Lee, C. C. J., Anton, M., Poon, C. S., & McRae, G. J. (2009). A kinetic model unifying presynaptic short-term facilitation and depression. *Journal of computational neuroscience, 26,* 459-473.

Hasani, R., Lechner, M., Amini, A., Rus, D., & Grosu, R. (2021, May). Liquid time-constant networks. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 35, No. 9, pp. 7657-7666).