Transmission Neural Networks[1]

From Virus Spread Models to Neural Networks

Shuang Gao

Research Fellow @ Simons Institute, UC Berkeley
shuang.gao@berkeley.edu

Workshop on Epidemics and Information Diffusion

October 28, 2022

Joint work with Peter E. Caines

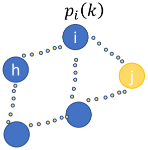
[1] Preprint available: arXiv:2208.03616

Outline

1. Motivation and Background
2. Virus Spread on Networks
3. Transmission Neural Networks (TransNNs)
4. TransNNs as Virus Spread Models
5. TransNNs as Learning Models
6. Conclusion and Future Work

Motivation: Virus Spread on Networks

Local graph structures are important for modelling the virus spread.



- Contact tracing
- Ring vaccination
- Covid exposure notification systems (bluetooth, location-based check-in, etc.)
- Computer virus spread

The underlying transmission network is crucial to monitor/predict/prevent virus spread.

Related Work: Virus Spread on Networks

- Epidemic model with heterogeneous transmissions [Lajmanovich and Yorke, 1976]
- **Discrete-time virus spread on given networks**: [Wang et al., 2003; Chakrabarti et al., 2008]
- Mean-field approximation for virus spread on networks: [Van Mieghem et al., 2008; Cator and Van Mieghem, 2012; Ferreira et al., 2012; Van Mieghem and van de Bovenkamp, 2015]
- Virus spread with network (structural) models: Random graphs [Kephart and White, 1992], Small-world [Moore and Newman, 2000], Degree distributions [Pastor-Satorras and Vespignani, 2001] ...
- Message-passing methods (influential nodes and control): [Karrer and Newman, 2010; Altarelli et al., 2014; Morone and Makse, 2015]
- Overview: Pastor-Satorras et al. [2015]; Nowzari et al. [2016]; Paré et al. [2020]; Kiss et al. [2017]

Outline

1. Motivation and Background
2. Virus Spread on Networks
3. Transmission Neural Networks (TransNNs)
4. TransNNs as Virus Spread Models
5. TransNNs as Learning Models
6. Conclusion and Future Work

Virus Spread on Effective Transmission Networks

Effective Transmission Network

Effective transmission link $i \to j$:
Virus passes from person i to person j and causes the infection of person j

Effective transmission network: network of persons with effective transmission links

Probability of Infection:

$$p_i(k) \triangleq \Pr(\text{Node } i \text{ is infected at time } k), \quad i \in [n].$$

One-step prediction:

$$(1 - p_i(k+1)) = \prod_{j \in N_i^\circ} (1 - p_j(k)), \quad i \in [n].$$

$N_i^\circ \triangleq \{j : (i,j) \in E\}$ denotes the neighbourhood of node i **with itself included**.

Virus Spread on Effective Transmission Networks

Nodal State via Shannon Information

Nodal state (Shannon Information):

$$s_i(k) \triangleq -\log(1 - p_i(k)) \in [0, +\infty]. \quad (1)$$

The state transformation $T(x) = -\log(1 - x)$ is monotone, bijective, and concave.

$$(1 - p_i(k+1)) = \prod_{j \in N_i^\circ} (1 - p_j(k)), \quad i \in [n]. \quad (2)$$

Linear dynamics under Shannon information states:

$$s_i(k+1) = \sum_{j \in N_i^\circ} s_j(k), \quad s_i(k) \in [0, +\infty], \;\; k \in \{0, 1, ...\}.$$

Virus Spread on Effective Transmission Networks

Explicit Solutions

Linear dynamics under Shannon information states:

$$s_i(k+1) = \sum_{j \in N_i^\circ} s_j(k) = \sum_{i=1}^{n} a_{ij} s_j(k), \quad s_i(k) \in [0, +\infty], \ \ k \in \{0, 1, ...\}.$$

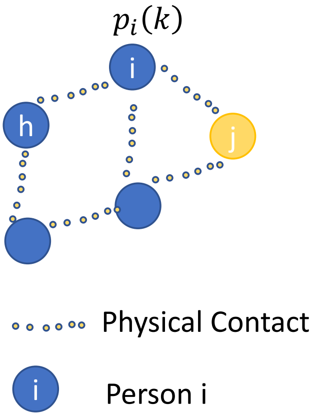
Let $s(k) = [s_1(k), ..., s_n(k)]^\mathsf{T}$ and $A = [a_{ij}]$ be the adjacency matrix with self-loops. Then $s(k) = A^k s(0)$ and we obtain

$$p_i(k) = 1 - e^{-[A^k s(0)]_i}, \quad i \in [n]$$

via the relation $s_i(k) \triangleq -\log(1 - p_i(k)) \in [0, +\infty]$.

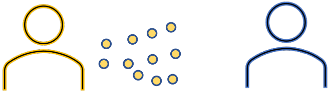
Linear dynamics and explicit solutions!

Virus Spread on Probabilistic Transmission Networks



$p_i(k) \triangleq$ probability of node i being infected at time k

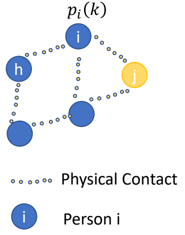
Multiple virus particles are transmitted across each link.



- a_{ij}: number of virus particles sent into the common space
- w_{ij}: probability of an **effective reception** of each virus particle sent from node j to node i

Virus Spread on Probabilistic Transmission Networks

Dynamics



Virus transmission model on networks[2] with heterogenous transmissions

$$1 - p_i(k+1) = \prod_{j \in N_i^\circ} \left(1 - w_{ij} p_j(k)\right)^{a_{ij}}, \quad i \in [n],\ k \in \{0, 1, ...\}$$

- $p_i(k)$: probability of being infected at time k
- a_{ij}: number of virus particles sent into the common space
- w_{ij}: probability of an effective reception of each virus particle from node j by node i
- N_i°: neighbourhood of node i on the physical contact network (including node i)

Assumption: Independences (in states and transmissions).

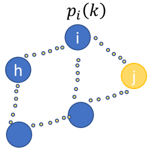
[2] Homogenous transmission probability (i.e. $w_{ij} = w$): Wang et al. [2003] and Chakrabarti et al. [2008]

Spread Process on Probabilistic Networks

Model Interpretations

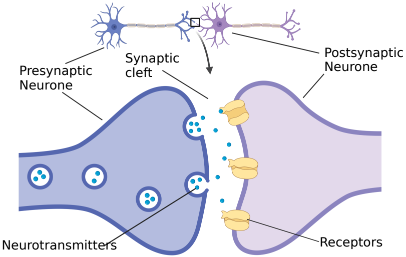
Characterizing dynamics: Activation by (only) one of the neighbors

$$1 - p_i(k+1) = \prod_{j \in N_i^\circ} \left(1 - w_{ij} p_j(k)\right)^{a_{ij}}$$



Different meanings of p_i, a_{ij}, w_{ij} leads to different interpretations:

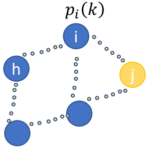
- Individual-level virus spread (e.g. contact network)
- Population-level virus spread (e.g. travel flow among cities)
- Information spread or opinion dynamics (e.g. social network)
- Neuronal network models (at neurotransmitter level)



Outline

1. Motivation and Background
2. Virus Spread on Networks
3. Transmission Neural Networks (TransNNs)
4. TransNNs as Virus Spread Models
5. TransNNs as Learning Models
6. Conclusion and Future Work

Transmission Neural Networks



Spread Process on Probabilistic Networks:

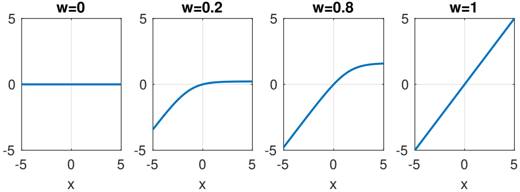
$$1 - p_i(k+1) = \prod_{j \in N_i^\circ} \left(1 - w_{ij} p_j(k)\right)^{a_{ij}}$$

via State Transformation (monotone, bijective, concave):

$$s_i(k) = -\log(1 - p_i(k)), \quad s_i(k) \in [0, +\infty] \text{ (Shannon Information)}$$

Transmission Neural Network (TransNN):

$$s_i(k+1) = \sum_{j=1}^{n} a_{ij} \Psi(w_{ij}, s_j(k))$$

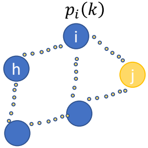


TLogSigmoid Activation Function

$$\Psi(w, x) \triangleq -\log\left(1 - w + w e^{-x}\right), \quad w \in [0, 1]$$

Transmission Neural Networks

Properties of TLogSigmoid Activation



Transmission Neural Network (TransNN):

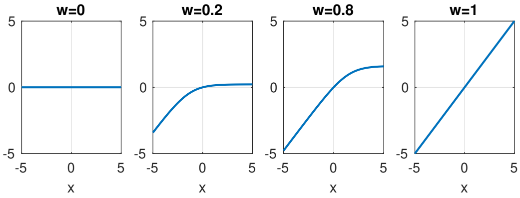
$$s_i(k+1) = \sum_{j=1}^{n} a_{ij}\Psi(w_{ij}, s_j(k)), \quad i \in [n],\ k \in \{0, 1, ...\}$$

TLogSigmoid Activation Function:

$$\Psi(w, x) \triangleq -\log\left(1 - w + we^{-x}\right), \quad w \in [0, 1]$$

Nice Properties of $\Psi(w, x)$:

- (a) concave in x
- (b) explicit derivatives (e.g. $\partial_x \Psi$, $\partial_w \Psi$ ...)
- (c) tuneable activation level $w \in [0, 1]$.



TransNN : $s_i(k+1) = \sum_{j=1}^{n} a_{ij}\Psi(w_{ij}, s_j(k))$, where $\Psi(w_{ij}, s_j) \triangleq -\log\left(1 - w_{ij} + w_{ij}e^{-s_j}\right)$

Connections with Standard Neural Networks

- Homogenous $w_{ij} = w$ and "activated" state $y_i(k) = \Psi(w, s_i(k)) \triangleq \sigma_w(s_i(k))$

$$\text{Standard NN Unit:} \quad y_i(k+1) = \sigma_w\left(\sum_{j=1}^{n} a_{ij} y_j(k)\right)$$

- Specializing to $w = 0.5$, TLogSigmoid activation becomes

$$\Psi(0.5, x) = \log\left(\frac{1}{1+e^{-x}}\right) + \log 2,$$

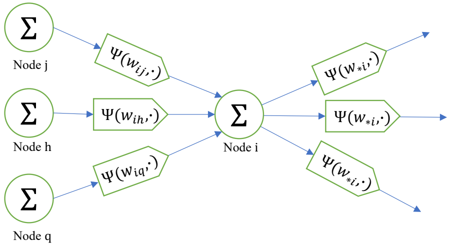
that is, LogSigmoid activation function with constant offset.

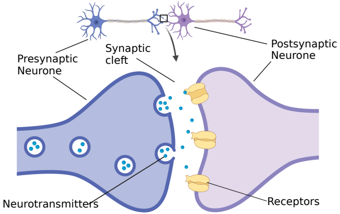
Transmission Neural Networks: Link Activation and Nonlinearity

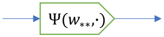
$$1 - p_i(k+1) = \prod_{q \in N_i^\circ} \left(1 - w_{ij} p_j(k)\right)^{a_{ij}}$$

is equivalent to

$$s_i(k+1) = \sum_{j=1}^{n} a_{ij} \Psi(w_{ij}, s_j(k))$$





Connection: TLogSigmoid 

$$\Psi(w_{ji}, s_i) = -\log(1 - w_{ji}(1 - e^{-s_i}))$$

Nodal State: $s_i = -\log(1 - p_i)$

Nodal Operation: Summation 

Transmission Neural Networks: Tuneable/Trainable Activation Func.

With state transformation $s_i = -\log(1 - p_i)$

$$s_i(k+1) = \sum_{j=1}^{n} a_{ij} \Psi(w_{ij}, s_j(k))$$

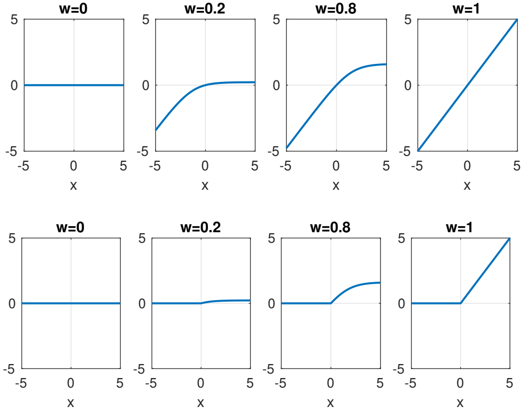
(1) **Tuneable LogSigmoid**:

$$\Psi(w, x) \triangleq -\log\left(1 - w + we^{-x}\right), \quad w \in [0, 1]$$

(2) **Tuneable LogSigmoid+** : (extending ReLU)

$$\Psi_+(w, x) \triangleq \begin{cases} \Psi(w, x), & x \geq 0 \\ 0, & x < 0 \end{cases}$$

when restricting the output $s_i = -\log(1 - p_i)$ to be non-negative.



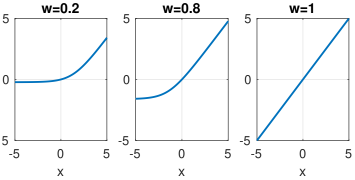
Transmission Neural Networks: Tuneable/Trainable Activation Func.

When taking state transformation: $s_i = \log(1 - p_i)$,

$$s_i(k+1) = \sum_{j=1}^{n} a_{ij} \Phi(w_{ij}, s_j(k))$$

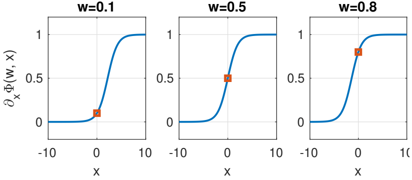
(3) **Tuneable SoftAffine**: (extending SoftPlus)

$$\Phi(w, x) \triangleq -\Psi(w, -x) = \log\left(1 - w + we^{x}\right)$$



(4) **Tuneable Sigmoid**: (extending Sigmoid)

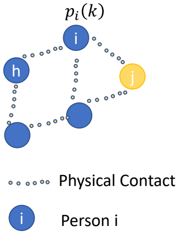
$$\partial_x \Phi(w, x) \triangleq \frac{we^{x}}{1 - w + we^{x}}$$



Outline

1. Motivation and Background
2. Virus Spread on Networks
3. Transmission Neural Networks (TransNNs)
4. TransNNs as Virus Spread Models
5. TransNNs as Learning Models
6. Conclusion and Future Work

TransNN as Virus Spread Model: Threshold Condition



Infection prob. over time steps:

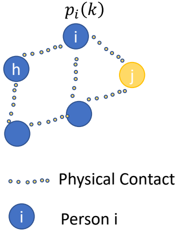
$$p(0) \to p(1) \to \dots \to p(k) \to \dots \stackrel{?}{\to} 0$$

The virus spread (probabilities) will die out regardless of initial conditions if

$$\max_{i \in [n]} |\lambda_i(A \odot W)| < 1, \quad \text{where } A \odot W = [a_{ij} w_{ij}]$$

and $\{\lambda_i(A \odot W) | i \in [n]\}$ denote all the eigenvalues of $A \odot W$. (see Thm. 1 GC 22')

TransNN as Virus Spread Model: Threshold Condition



Proof Idea (one direction): Concavity of $\Psi(w,x)$ in $x \in [-\infty,+\infty]$ implies that

$$\Psi(w,z) \le \Psi(w,x) + \partial_x \Psi(w,x)(z-x), \quad \forall x,z \in [-\infty,+\infty].$$

Applying this property to the virus spread model yields

$$s_i(k+1) \le \sum_{j=1}^{n} a_{ij}\left(\Psi(w_{ij}, s_j^*) + \partial_x \Psi(w_{ij}, s_j^*)(s_j(k) - s_j^*)\right).$$

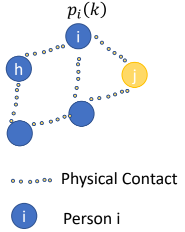
Choosing $s^* = 0$ yields

$$s_i(k+1) \le \sum_{j=1}^{n} a_{ij} w_{ij} s_j(k), \quad i \in [n].$$

Discrete time linear system $x(k+1) = [A \odot W]x(k)$ is (globally asymptotically) stable iff

$$\max_{i\in[n]} |\lambda_i(A \odot W)| < 1, \quad \text{where } A \odot W = [a_{ij}w_{ij}].$$

Epidemic Threshold Condition: Special Case



Threshold Condition:

$$\max_{i \in [n]} |\lambda_i(A \odot W)| < 1, \quad \text{where} A \odot W = [a_{ij} w_{ij}]$$

Special Case:

When $w_{ii} = 1 - \delta$ and $w_{ij} = \beta$, $i \neq j$, with δ as the recover probability and β as the infection probability,

$$A \odot W = \beta A + I(1 - \delta - \beta).$$

Then it is equivalent to the well-known threshold condition[3]:

$$\lambda_{\max}(\tilde{A}) < \frac{\delta}{\beta}, \quad \text{where } \tilde{A} \triangleq A - I.$$

[3] See Chakrabarti et al. [2008]

TransNN as Virus Spread Model: Continous Time TransNNs

Discrete Time TransNN : $s_i(k+1) = \sum_{j=1}^{n} a_{ij}\Psi(w_{ij}, s_j(k)), \quad \Psi(w,s) \triangleq -\log\left(1 - w + we^{-s}\right)$

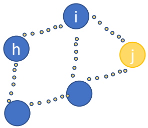
Extra Assumptions on Transmission Probability w.r.t. time duration Δ:

$$w_{ij} = c_{ij}\Delta + o(\Delta), \quad i \neq j$$
$$w_{ii} = 1 - c_{ii}\Delta + o(\Delta), \quad (\text{e.g. } w_{ii} = e^{-c_{ii}\Delta})$$

$c_{ij} \geq 0$ as basic transmission probability rate (per unit time) from j to i
$c_{ii} \geq 0$ as self-healing probability rate (per unit time)

Continous Time TransNN : $\frac{ds_i(t)}{dt} = \sum_{j \in N_i^o, j \neq i} a_{ij}c_{ij}(1 - e^{-s_j(t)}) + c_{ii}(1 - e^{s_i(t)})$

Continous Time TransNNs is Equivalent to Network SIS



Extra Assumptions on Transmission Probability w_{ij}:

$$w_{ij} = c_{ij}\Delta + o(\Delta), \quad \text{with time duration } \Delta$$
$$w_{ii} = e^{-c_{ii}\Delta} = 1 - c_{ii}\Delta + o(\Delta), \quad \forall i, j \in [n], i \neq j,$$

Continous Time TransNN :
$$\frac{ds_i(t)}{dt} = \sum_{j \in N_i^o, j \neq i} a_{ij} c_{ij} (1 - e^{-s_j(t)}) + c_{ii}(1 - e^{s_i(t)})$$

via $s_i(t) = -\log(1 - p_i(t))$, is equivalent to

Continous Time Network SIS[4] :
$$\frac{dp_i(t)}{dt} = (1 - p_i(t)) \sum_{j \in N_i^o, j \neq i} a_{ij} c_{ij} p_j(t) - c_{ii} p_i(t).$$

[4] Proposed and developed by Lajmanovich and Yorke [1976]; Van Mieghem et al. [2008]

TransNNs Summary: Discrete-Time vs Continous-Time

Discrete Time Virus Spread

$$1 - p_i^+ = \prod_{j \in N_i^o} (1 - w_{ij} p_j)^{a_{ij}}$$

(A1) $\Delta \to 0$

Continous Time Virus Spread

$$\dot{p}_i = (1 - p_i) \sum_{j \neq i} a_{ij} c_{ij} p_j - c_{ii} p_i$$

$$s_i = -\log(1 - p_i)$$

$$s_i = -\log(1 - p_i)$$

Discrete Time TransNNs

$$s_i^+ = \sum_{j \in N_i^o} a_{ij} \Psi(w_{ij}, s_j)$$

(A1) $\Delta \to 0$

Continous Time TransNNs

$$\dot{s}_i = \sum_{j \neq i} a_{ij} c_{ij} (1 - e^{-s_j}) + c_{ii} (1 - e^{s_i})$$

(A1) Assumption:

$$w_{ij} = c_{ij}\Delta + o(\Delta)$$
$$w_{ii} = 1 - c_{ii}\Delta + o(\Delta)$$

Outline

1 Motivation and Background
2 Virus Spread on Networks
3 Transmission Neural Networks (TransNNs)
4 TransNNs as Virus Spread Models
5 TransNNs as Learning Models
6 Conclusion and Future Work

TranNNs as Learning Models

Universal Function Approximator

Definition (Universal Function Approximator[5])

A set $\mathcal{M}$ of (parameterized) functions in $L^\infty_{loc}(\mathbb{R}^d;\mathbb{R}^m)$ is called a *Universal Function Approximator for* $C(\mathbb{R}^d;\mathbb{R}^m)$ if given any $\varepsilon > 0$, any compact subset of $K \subseteq \mathbb{R}^d$ and any $f \in C(K;\mathbb{R}^m)$, there exists $F \in \mathcal{M}$ such that

$$\operatorname*{ess\,sup}_{x \in K} \|F(x) - f(x)\| < \varepsilon.$$

In other words, $\mathcal{M}$ is a universal function approximator for $C(\mathbb{R}^d;\mathbb{R}^m)$ if it is *dense* in $C(\mathbb{R}^d;\mathbb{R}^m)$ in the topology of uniform convergence on compacta.

[5] Pinkus [1999]; Leshno et al. [1993]; Hornik et al. [1989]

Universal Function Approximator

TransNNs with One Hidden Layer

Input: $x \in \mathbb{R}^d$

Output: $y^\theta(x) \in \mathbb{R}$

$$y^\theta(x) = \sum_{i=1}^{n} a_i \Psi(w_i, \eta_i^\top x + b)$$

TLogSigmoid Activation:

$$\Psi(w, x) \triangleq -\log\left(1 - w + we^{-x}\right)$$

Fixed Bias $b \neq 0$.

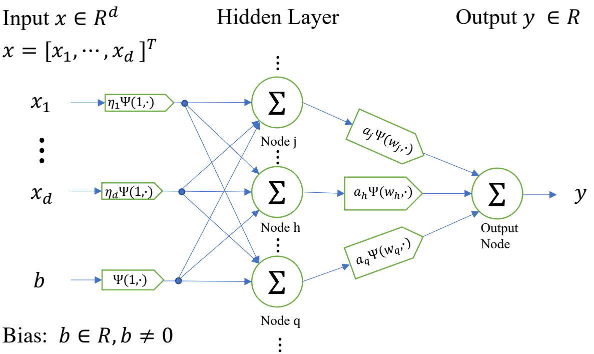


Figure: TransNN with one hidden layer. We note that $\Psi(1, \alpha) = \alpha$ for $\alpha \in \mathbb{R}$.

Universal Function Approximator (cont.)

TransNNs with One Hidden Layer

Theorem (Universal Function Approximator)

TransNN with one hidden layer, a ***fixed bias term*** $b \neq 0$ *and* ***rational weights*** $\{a_i\}$ *as*

$$y^\theta(x) = \sum_{i=1}^{n} a_i \Psi(w_i, \eta_i^\intercal x + b), \quad x \in \mathbb{R}^d,\ y^\theta(x) \in \mathbb{R} \tag{4}$$

with arbitrary parameters $\theta \triangleq (n, (a_i)_{i=1}^n, (\eta_i)_{i=1}^n, (w_i)_{i=1}^n)$ *in* $\Theta_{\mathbb{Q}}$*, is a* ***Universal Function Approximator***[6] ***for*** $C(\mathbb{R}^d)$*, where*

$$\Theta_{\mathbb{Q}} \triangleq \Big\{(n, (a_i)_{i=1}^n, (\eta_i)_{i=1}^n, (w_i)_{i=1}^n) \Big| n \in \mathbb{N},\ a_i \in \mathbb{Q},\ \eta_i \in \mathbb{R}^d,\ w_i \in [0,1]\Big\}.$$

Proof follows closely that of [Leshno et al., 1993, Theorem 1].

[6] That is, the set of functions characterized by TransNNs with parameters in Θ_Q is dense in $C(\mathbb{R}^d; \mathbb{R})$ in the topology of uniform convergence on compacta.

TransNNs as Learning Models: Feedforward NN Examples

TransNN: $$s_i(k+1) = \sum_{j=1}^{n} a_{ij}^k \Psi(w_{ij}^k, s_j(k)), \quad i \in [n], k \in \{0, 1, 2..., T-1\}$$

Input: $s(0) \triangleq [s_1(0), ..., s_n(0)]^\intercal$ Output: $s(T) \triangleq [s_1(T), ..., s_n(T)]^\intercal$. That is

$$s(T) = \text{TransNN}_\theta(s(0))$$

Learning objective with data $\{(x^{(i)}, y^{(i)})\}_{i=1}^{D}$:

$$\min_{\theta \in \Theta} \left\{ \frac{1}{D} \sum_{i=1}^{D} l \left(\text{obs}(\text{TransNN}_\theta(x^{(i)})),\ y^{(i)} \right) + r(\theta) \right\}$$

where $l(\cdot, \cdot)$: loss function $r(\theta)$: regularization Θ : all feasible parameters

Example of output observation : $p = 1 - \exp_\circ(-s) \triangleq \text{obs}(s)$.

TransNNs as Learning Models: Examples

TransNN:

$$s_i(k+1) = \sum_{j=1}^{n} a_{ij}^k \Psi(w_{ij}^k, s_j(k)), \quad i \in [n], k \in \{0, 1, 2..., T-1\}$$

For Recurrent Neural Networks, Graph Neural Networks and others:

- use TLogSigmoid, TLogSigmoid+ or TSoftAffine activations.
- take sum of "link-activated states"

TransNNs as Learning Models: Advantages

Advantages of using TransNN as Learning Models:

- Interpretability:
  Using TLogSigmoid, TLogSigmoid+ or TSoftAffine activations functions, yields the natural interpretation of **Probabilities of nodes being active**!
- Automatic Selection of Activations:
  Automatic selection of a set of activation functions
  (including ReLU, SoftPlus, LogSigmoid as special cases)
- Activations with Links:
  (a) Link activation levels
  (b) Learnable activation levels with fixed graph structures

Conclusion and Future Work

Conclusion

- TransNNs as Virus Spread Models
  - (a) Threshold conditions
  - (b) Linking discrete-time and continous-time SIS models on networks

Conclusion and Future Work

Conclusion

- TransNNs as Virus Spread Models
  - (a) Threshold conditions
  - (b) Linking discrete-time and continous-time SIS models on networks
- TransNNs as Learning Models
  - (a) Universal function approximator
  - (b) Tuneable activation functions (TLogSigmoid, TLogSigmoid+, TSoftPlus, TSigmoid)
  - (c) Automatic selection of activation functions
  - (d) Interpretations of activation probabilities!

Conclusion and Future Work

Future Work

- Control and modulation of TransNNs (in both epidemics and learning)

Conclusion and Future Work

Control Variables for TransNNs as Virus Spread Models: $s_i^+ = \sum_{j=1}^n a_{ij}\Psi(w_{ij}, s_j)$

Individual perspective or social planner perspective

1. Wearing mask:
   (by reducing $u_i w_{ij}$ and $a_{ij} v_j$ where u_i, v_i denote the inward and outward effectiveness of wearing masks)
2. Social distancing:
   (by reducing a_{ij}, e.g. $a_{ij} e^{-r_{ij}^2}$ where r_{ij} is the distance)
3. Vaccination:
   (by reducing $v_i w_{ij}$ where v_i denotes the effectiveness of vaccination)
4. Treatment:
   (by reducing $w_{ii} = 1 - \tau_i \delta_i$ via increasing the recovery probability $\tau_i \delta_i$ where τ_i denotes the effectiveness of treatment)

Global Modulation: $w_{ij} = \gamma \omega_{ij}$

Conclusion and Future Work

Future Work

- Control and modulation of TransNNs (in both epidemics and learning)
- Random realizations of (1) connections and (2) states (in epidemics and learning)
- TransNNs with inhibitions and plasticity motivated by biological neuronal networks

Conclusion and Future Work

Future Work

- Control and modulation of TransNNs (in both epidemics and learning)
- Random realizations of (1) connections and (2) states (in epidemics and learning)
- TransNNs with inhibitions and plasticity motivated by biological neuronal networks
- Training TransNNs to estimate and predict virus spread
  (respecting local structures, based on partial historical observations)
- Derivation of epidemic models on networks with more nodal states and extra features
  (such as location and age) based on TransNNs
- ...

Conclusion and Future Work

Thank you!

References

Fabrizio Altarelli, Alfredo Braunstein, Luca Dall'Asta, Joseph Rushton Wakeling, and Riccardo Zecchina. Containing epidemic outbreaks by message-passing techniques. *Physical Review X*, 4(2):021024, 2014.

Eric Cator and Piet Van Mieghem. Second-order mean-field susceptible-infected-susceptible epidemic threshold. *Physical review E*, 85(5):056111, 2012.

Deepayan Chakrabarti, Yang Wang, Chenxi Wang, Jurij Leskovec, and Christos Faloutsos. Epidemic thresholds in real networks. *ACM Transactions on Information and System Security (TISSEC)*, 10(4):1–26, 2008.

Silvio C Ferreira, Claudio Castellano, and Romualdo Pastor-Satorras. Epidemic thresholds of the susceptible-infected-susceptible model on networks: A comparison of numerical and theoretical results. *Physical Review E*, 86(4):041125, 2012.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

Brian Karrer and Mark EJ Newman. Message passing approach for general epidemic models. *Physical Review E*, 82(1):016101, 2010.

Jeffrey O Kephart and Steve R White. Directed-graph epidemiological models of computer viruses. In *Computation: the micro and the macro view*, pages 71–102. World Scientific, 1992.

István Z Kiss, Joel C Miller, Péter L Simon, et al. Mathematics of epidemics on networks. *Cham: Springer*, 598:31, 2017.

Ana Lajmanovich and James A Yorke. A deterministic model for gonorrhea in a nonhomogeneous population. *Mathematical Biosciences*, 28(3-4):221–236, 1976.

Moshe Leshno, Vladimir Ya Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural networks*, 6(6):861–867, 1993.

Cristopher Moore and Mark EJ Newman. Epidemics and percolation in small-world networks. *Physical Review E*, 61(5):5678, 2000.

Flaviano Morone and Hernán A Makse. Influence maximization in complex networks through optimal percolation. *Nature*, 524(7563):65–68, 2015.

Cameron Nowzari, Victor M Preciado, and George J Pappas. Analysis and control of epidemics: A survey of spreading processes on complex networks. *IEEE Control Systems Magazine*, 36(1):26–46, 2016.

Philip E Paré, Carolyn L Beck, and Tamer Başar. Modeling, estimation, and analysis of epidemics over networks: An overview. *Annual Reviews in Control*, 50:345–360, 2020.

Romualdo Pastor-Satorras and Alessandro Vespignani. Epidemic spreading in scale-free networks. *Physical Review Letters*, 86(14):3200, 2001.

Romualdo Pastor-Satorras, Claudio Castellano, Piet Van Mieghem, and Alessandro Vespignani. Epidemic processes in complex networks. *Reviews of Modern Physics*, 87(3):925, 2015.

Allan Pinkus. Approximation theory of the MLP model in neural networks. *Acta Numerica*, 8:143–195, 1999.

P Van Mieghem and Ruud van de Bovenkamp. Accuracy criterion for the mean-field approximation in susceptible-infected-susceptible epidemics on networks. *Physical Review E*, 91 (3):032812, 2015.

Piet Van Mieghem, Jasmina Omic, and Robert Kooij. Virus spread in networks. *IEEE/ACM Transactions On Networking*, 17(1):1–14, 2008.

Yang Wang, Deepayan Chakrabarti, Chenxi Wang, and Christos Faloutsos. Epidemic spreading in real networks: An eigenvalue viewpoint. In *22nd International Symposium on Reliable Distributed Systems, 2003. Proceedings.*, pages 25–34. IEEE, 2003.