Bayesian Decision Flow Diagrams

An Agent Based Modeling Technique for Combining Domain Expert Knowledge and Learning From Data

Parantapa Bhattacharya, Christopher Kuhlman, Samarth Swarup, Mandy Wilson, and Mark Orr

Network System Science and Advanced Computing
Biocomplexity Institute and Initiative
University of Virginia

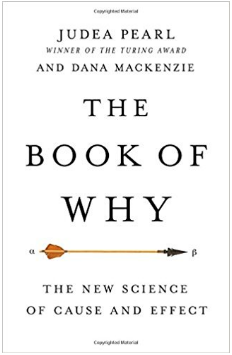
ICTS - TBML 2018

Bayesian Decision Flow Diagrams (BDFD)

- A graphical method for describing complex agent-based models
- Combines:
  - Modeling prowess of Bayesian networks
  - Expressiveness of flow charts/flow diagrams
- Can utilize both:
  - Domain expert knowledge for system modeling
  - Learning individual decision models from data
- Can be used to express causal assumptions
  - Facilitates explainability of observed data
  - Facilitates answering of counterfactual queries

Analysis and Modeling for Causality

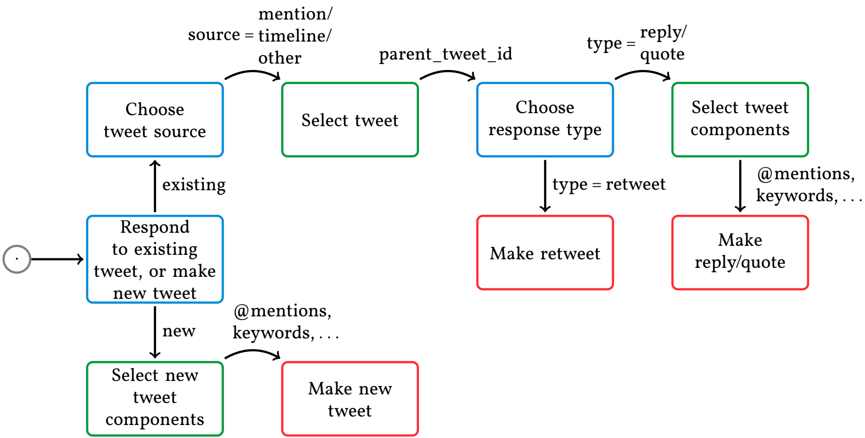
- Key objective for modeling
  - Answering “why”, questions of explainability
  - Answering “what-if”, counter factual questions
  - Answering causal questions
- No matter the magnitude of observational data
  - “Correlation is not causation”
- Randomized control trails are not always feasible
- Causal models are the next best alternative



The Twitter Simulation Problem

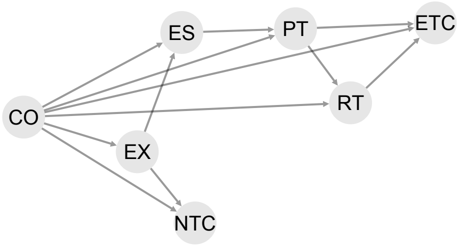
- Model a subpopulation of Twitter users
- Agents are Twitter users
- Discrete time simulation
- Every timestep users generate tweets
  - Make new tweet
  - Respond to existing tweet: retweet, reply, quote
- Output: An ordered sequence of tweets

Niobe: A Twitter BDFD model



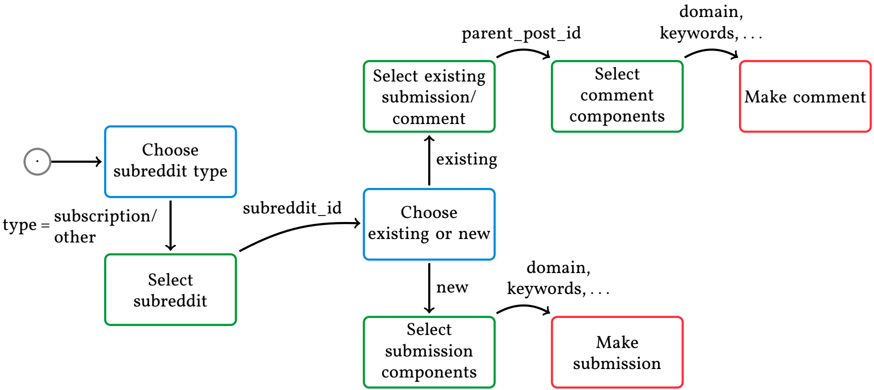
Context: user ID, current time, all previous tweets

Niobe: A Twitter BDFD model (BayesNet)



CO: Context, EX: Existing or new, ES: Existing tweet source
NTC: New tweet components, PT: Parent tweet, RT: Response type
ETC: Existing tweet response components

Persephone: A Reddit BDFD model



Context: user ID, current time, all previous submissions and comments

Learning in BDFD (Niobe) setup

- Training data generation
  - For given tweet, find paths that Niobe can take to generate it
  - For each path, compute likelihood of taking path
  - Randomly select one path, with probability proportional to likelihood
  - For every decision node on path, compute features and label
  - Append features and label to decision node's training data
- Individual model training
  - Train the individual models of decision nodes independently

Practical Concerns

- Scaling issues caused due to big data
- Dependence on context makes computation sequential
- Workaround
  - Control time granularity of simulation
  - Assumption: Agents can compute their actions independently, within a single timestep
  - Allows for bulk synchronous parallel task decomposition
- **The Matrix ABM platform**
  - Takes care of synchronization and orchestration

Bayesian Decision Flow Diagrams (BDFD)

- A graphical method for describing complex agent-based models
- Can utilize both:
  - Domain expert knowledge for system modeling
  - Learning individual decision models from data
- Can be used to express causal assumptions
- BDFD is still work in progress

Inference Problems for Graphical Dynamical Systems

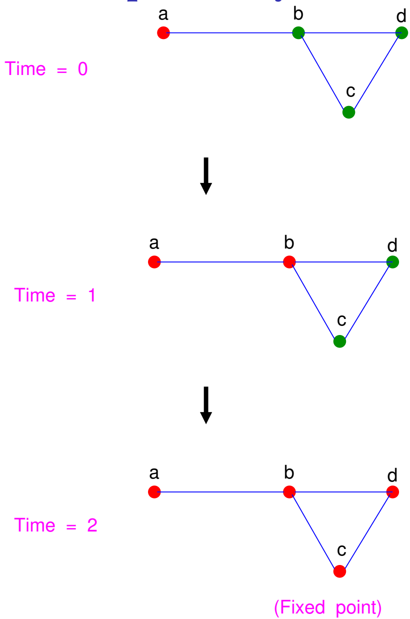
Graphical Dynamical Systems: Basics

Components of a **Graphical Dynamical System** $\mathcal{S}$:

- An underlying (undirected or directed) graph $G(V, E)$.
- **Nodes:** Agents in the network.
- **Edges:** Local interactions among the agents.
- A **state value** (0 or 1) for each node.
- A Boolean (deterministic or probabilistic) **local transition function** for each node specifying how the next state of a node depends on its current state and those of its neighbors.
- **Update mechanism: synchronous**. (Other possibilities: sequential, block sequential.)

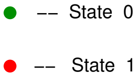
Terminology: Synchronous Dynamical System (SyDS)

An Example of a SyDS and its Time Evolution

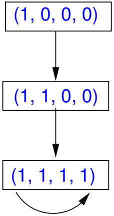


Each local function: OR

Notation:



System's trajectory:



Some Definitions

- **Configuration** at time t: Vector specifying the state of each node at time t.
- **Successor** of a configuration $\mathcal{C}$: The configuration that **immediately follows** $\mathcal{C}$ in time evolution.

Inference Problem: (General)
Given a **partially specified** SyDS S', infer the other parts of S'.

Inference problem considered: Given the underlying graph of a SyDS $\mathcal{S}$ and the concept class for the local functions, infer the local functions.

Inference Modes and Classes of Local Functions

Passive mode: Use observations (e.g., fixed points, partial trajectories) to carry out inference.

Active mode: Interact with the system through **queries** to carry out inference. (Each query q specifies a configuration and the system's response is the **successor** of q.)

Classes of local functions considered:

- Deterministic threshold functions
- Probabilistic threshold functions
- Edge probabilities in the independent cascade model

Representative Results

- Efficient algorithms for inferring deterministic threshold functions given a collection of trajectories.
  [Adiga et al., TCS 2017 and CIAA 2015].
- Efficient algorithms for inferring deterministic and probabilistic threshold functions using near-minimal number of queries.
  [Adiga et al., AAAI 2018 and Complex Networks 2018].
- An efficient algorithm for inferring the edge probabilities for the independent cascade model approximately using near-minimal number of queries. [Adiga et al., CIKM 2018]

List of Group's Papers on Inference

- A. Adiga et al., "Using Active Queries to Learn Local Stochastic Behaviors in Social Networks", *Proc. 7th Intl. Conf. on Complex Networks and Their Applications*, Dec. 2018.
- A. Adiga, et al., "Inferring Probabilistic Contagion Models Over Networks Using Active Queries", *Proc. CIKM 2018*, Oct. 2018, pp. 377–386.
- A. Adiga, et al., "Inferring Users' Choice Functions in Networked Social Systems Through Active Queries", *Conf. Notes, COMSOC-2018*, June 2018, 19 pages.
- A. Adiga et al., "Learning the Behavior of a Dynamical System Via a '20 Questions' Approach", *Proc. AAAI-2018*, Feb. 2018, pp. 4630–4637.
- A. Adiga, et al., "Inferring the Local Transition Functions of Discrete Dynamical Systems from Observations of System Behavior", *Theo. Comp. Sci.*, Vol. 679, 2017, pp. 126–144.
- A. Adiga, et al., "Complexity of Inferring Local Transition Functions of Discrete Dynamical Systems", *Proc. CIAA 2015*, Lecture Notes in CS, Vol. 9223, Aug. 2015, pp. 21–34.

Thanks!

parantapa@virginia.edu

The research is partially based upon work supported by the Defense Advanced Research Projects Agency (DARPA), via the Air Force Research Laboratory (AFRL). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA, the AFRL or the U.S. Government.