TNDCN: A Unified Framework for Representation Learning on Dynamic Social Interaction Networks

Pan Li
Computer Science, Stanford University

Joint work with
Y. Wang, J. Leskovec, C. Bai, VS. Subrahmanian.

Interaction Networks

- Interactions between people convey indicative patterns that reflect ones' social status and personal characters
  - interaction --- behaviors: A looks at/talks to/… B

- Interaction networks can be used for **modeling** and **downstream prediction tasks**…

But how?

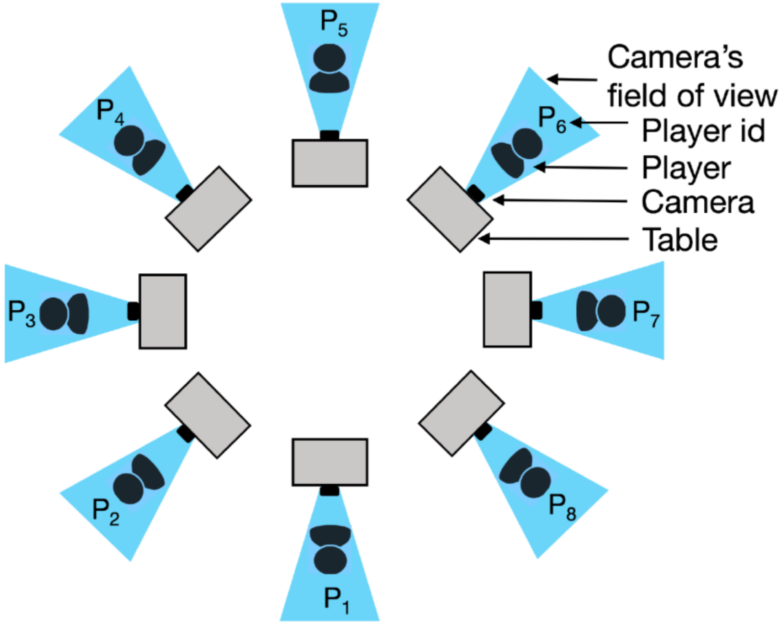
This talk:

Step 1: Review of network construction and basic analysis

Step 2: Network algorithm for predictions (TNDCN) social-related character prediction

The "mafia" game

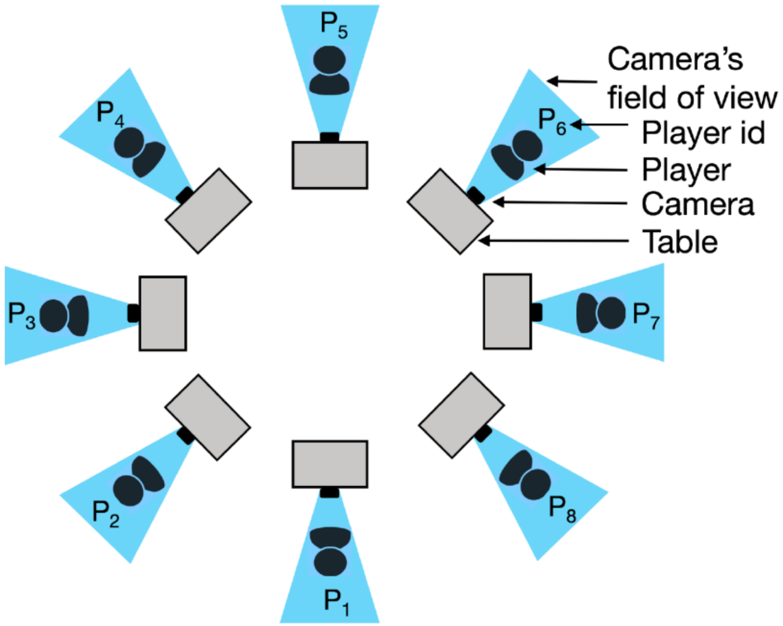
- A game of 8 players
- Players have assigned (but unknown) roles:
  - Truth-tellers
  - Deceivers
- In the end one of the two groups wins
- Surveys were conduct during the game to collect some characters.



Extracting Dynamic Networks :

- **Camera is set up**

- For **every 1/3rd second**, estimate every person's interaction with others from the video



Joint work with Dartmouth, UCSB, Arizona

Extracting Dynamic Networks :

- **ICAF:** a collective classification predictive model
- **We generate 62 networks:**
  - Publicly released the networks to promote future research: http://snap.stanford.edu/data/comm-f2f-Resistance.html

Dataset statistics	
Number of networks	62
Number of nodes	451
Number of edges	3,126,993
Average number of edges per network	50,435
Total temporal length	142,005 seconds
Average temporal length per network	2,290 seconds

Networks were used to evaluate our model!

Basic analysis of networks

- Deceivers tend to have unbalanced visual focus on others (low entropy).
- Deceivers ignore other Deceivers and interact more with Truth-Tellers.
- Deceivers speak less and are listened to less

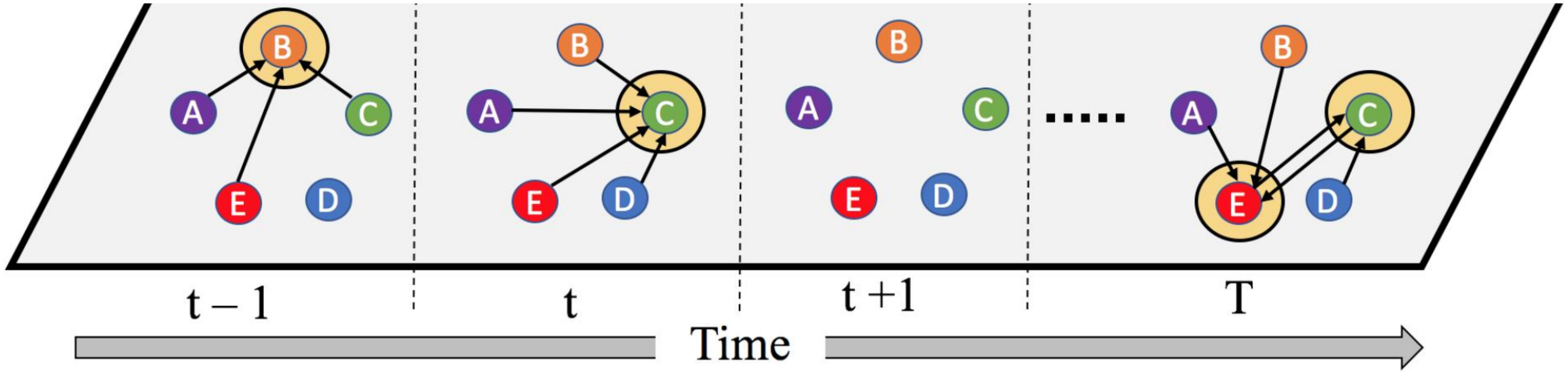
↓ discriminatory features

Can we identify who is a deceiver (or other types of characters) using the networks?

Model #1: Temporal Graph Convolution

- **Input:** Sequence of graph snapshots
- **Output:** Node labels
- **Steps:**

1. Run graph neural network model on each network
2. Aggregate outputs from the sequence of graphs
   - Aggregations: average, input to LSTM, input to RNN, etc.



Our Model #2: Propagation over Neg-Network

1. **Create a negative network for each 1 second fragment:**
   - Replace each edge weight $w_{i,j}$ with $1 - w_{i,j}$
2. **Initialize nodes using a feature vector:**
   - **Node features:** Fraction of speaking, entropy of looking at, in-degree, in-degree while speaking
3. **Run till convergence on each network**

$$S_i \leftarrow \beta * (\Sigma_{(i,j) \in E} S_j * (1 - w_{j,i})) + (1 - \beta) S_i$$

4. **Average S scores over all networks/time steps**

Avoiding eye-focus on others seems to be informative indicators of deceivers

Prediction Results

- **Task:** Identify who is a Deceiver
- **Setting:** 5-fold cross validation results:
  - Split by game to avoid label leakage: 80% games are used for training, 20% testing

Baselines

Method	Performance
Emotion	0.538 AUC
Head and eye movement	0.549 AUC
Facial action unit	0.569 AUC
Late fusion	0.587 AUC
Graph Neural Network	0.596 AUC
Propagation on negative network	**0.73 AUC**

Questions

- Our survey includes other labels (precious, labeled by human), such as dominance and nervousness. Can we predict them?

- Can we design a model uniformly good for different prediction tasks?

--- Not rely on the pre-design correlation analysis between features and labels

- Can the designed model hold certain self-explaining power to indicate important features?

feature analysis -> model design

general models -> important feature (-> feature analysis)

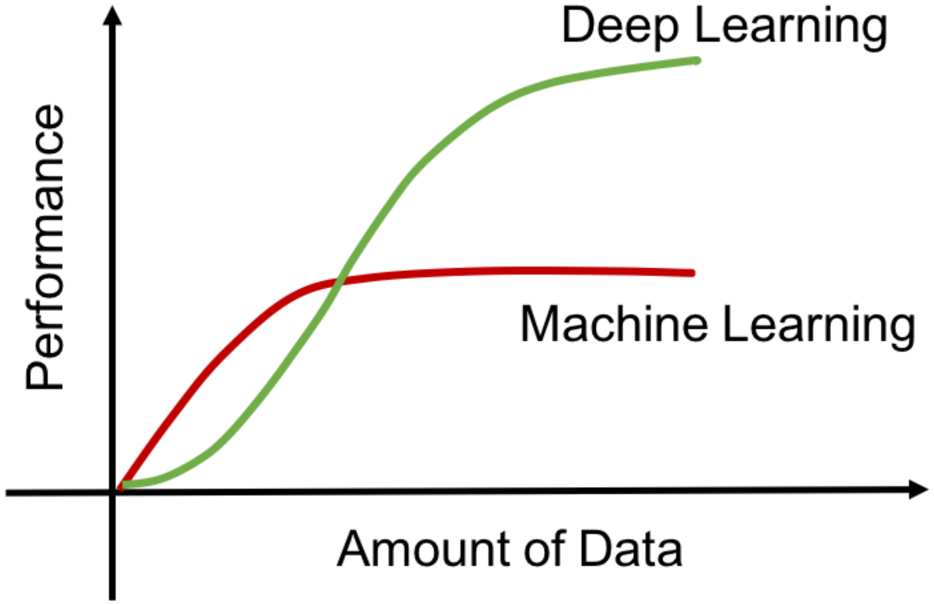
This talk:

Step 1: Review of network construction and basic analysis

Step 2: Network algorithm for predictions (TNDCN) social-related character prediction

Deep learning models

- Advantage I --- general: Deep models depend less on (task-specific) feature selection
- Advantage II --- expressive: Deep models capture complex patterns (hard for manually designed)
- Disadvantage --- data-hungry: Labels are scarce in our case and many other social-scientific use cases



Challenging! --- Previous results show that naïve using graph neural networks may not work well…

How about other proposed deep models?

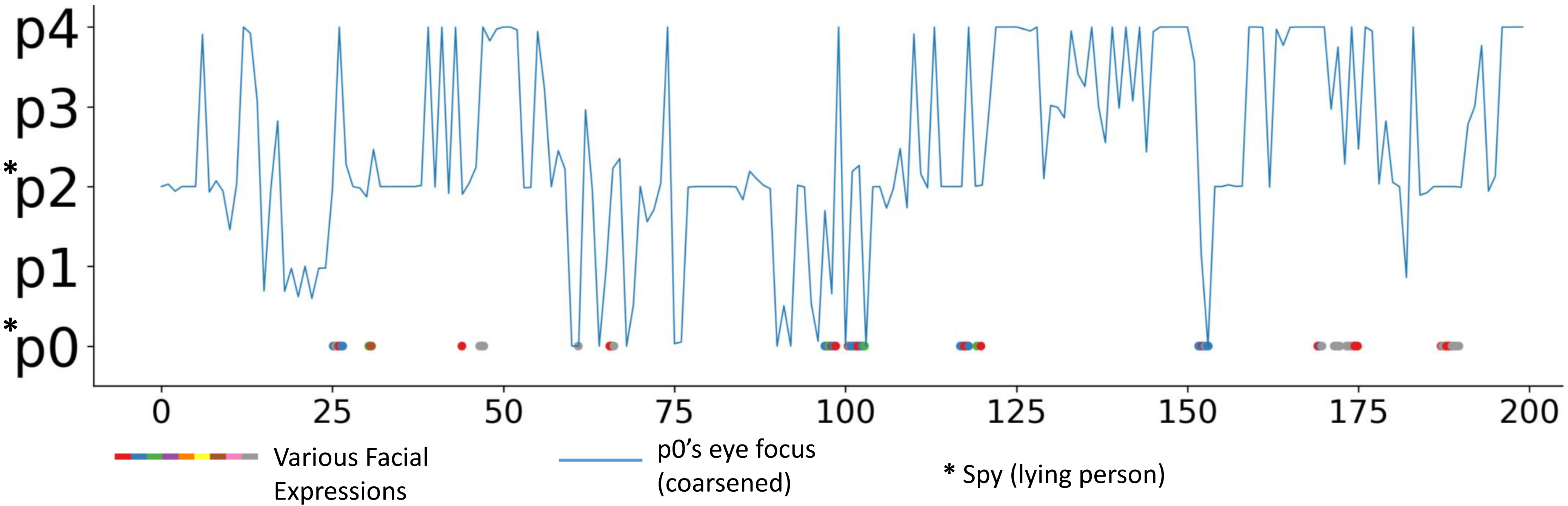
- General graph neural networks for dynamic networks (mostly for future-interaction prediction) are hard to be applied to --- deception detection, dominance identification, nervousness detection…

Challenge #1

• Interactions and node features (facial expressions, …) can be highly dynamic

--- People shift their eye focus 60+ times on average within one minute.

--- Interleaving patterns are subtle and scattered in a long time span (30+min/game)

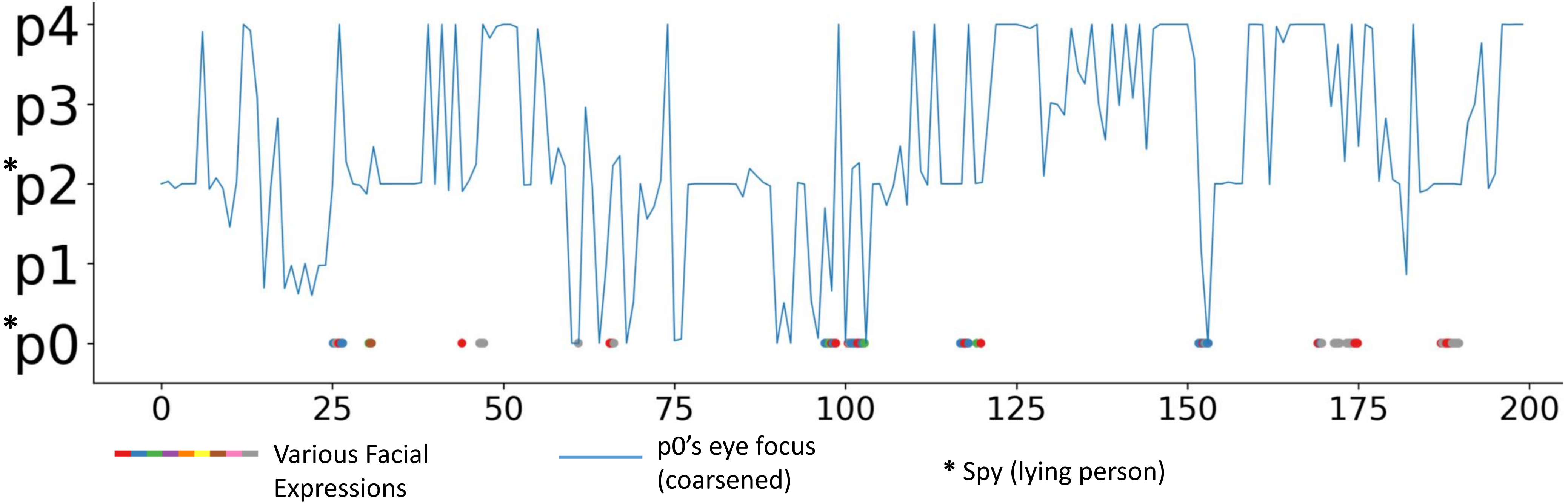


Challenge #2

• Interactions are <u>complex</u>

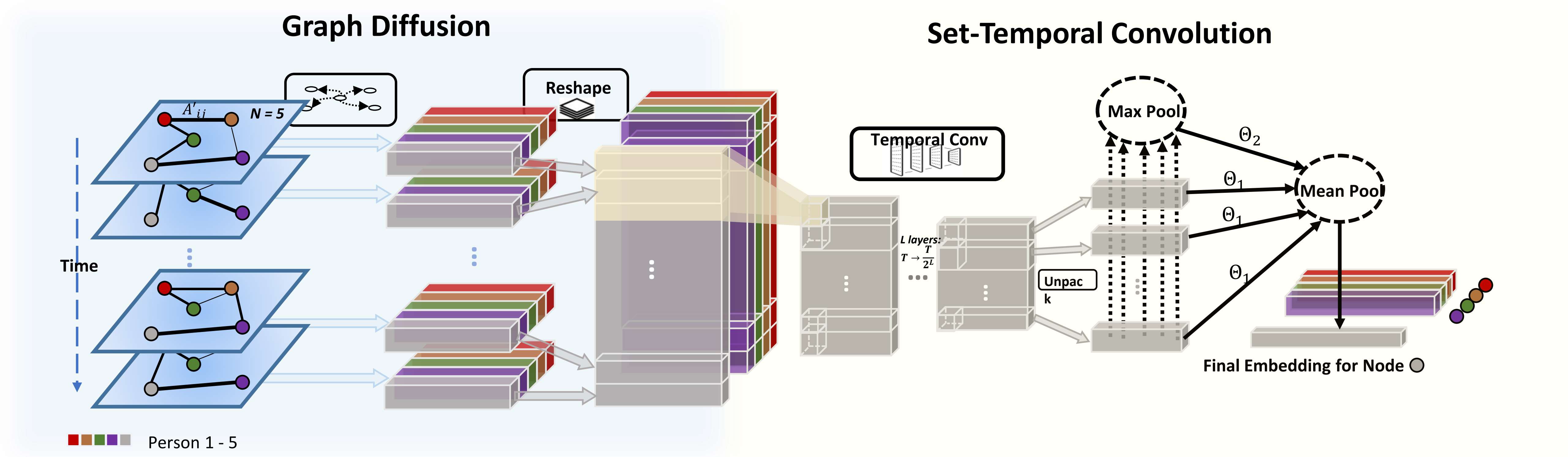
--- Interactions are with time-variant durations, concurrent

--- A complex model is needed but labels are scarce



* Spy (lying person)

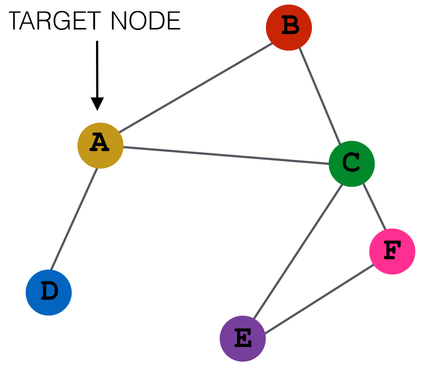
Temporal-Network Diffusion Convolution (TNDCN)



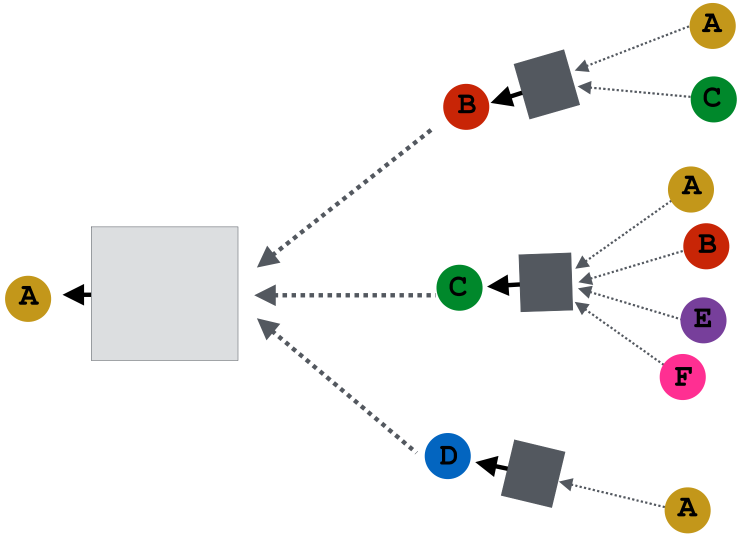
1. Graph diffusion refines node features in each network snapshot
2. Set-temporal convolution aggregates the refined node features over time

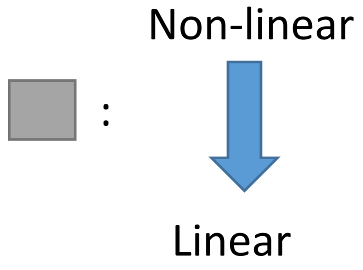
Graph Diffusion --- I

- Modeling long-hop interweaving of highly dynamic node attributes and interactions with only a few labels



INPUT GRAPH

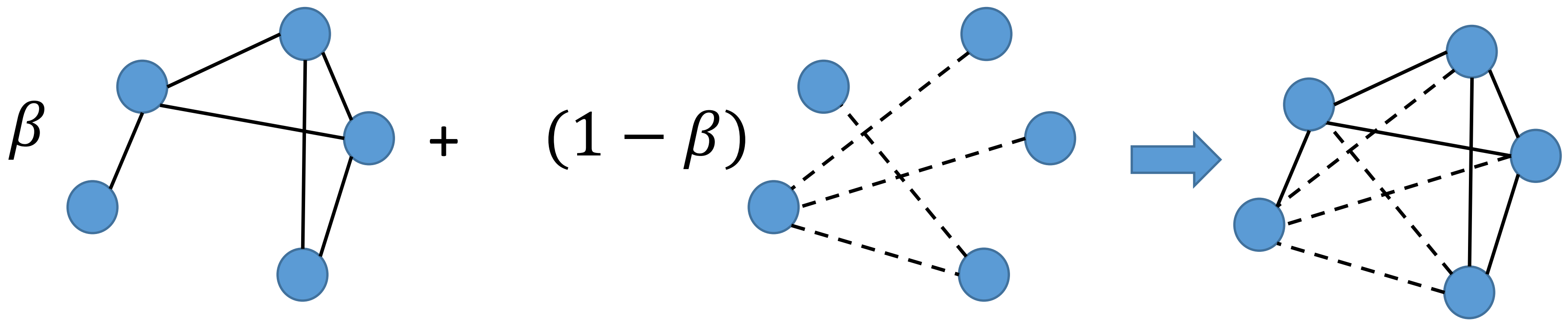




Less representative but much deeper with less parameters

Graph Diffusion --- II

- The input graph combines positive and negative graphs via a trainable parameter.



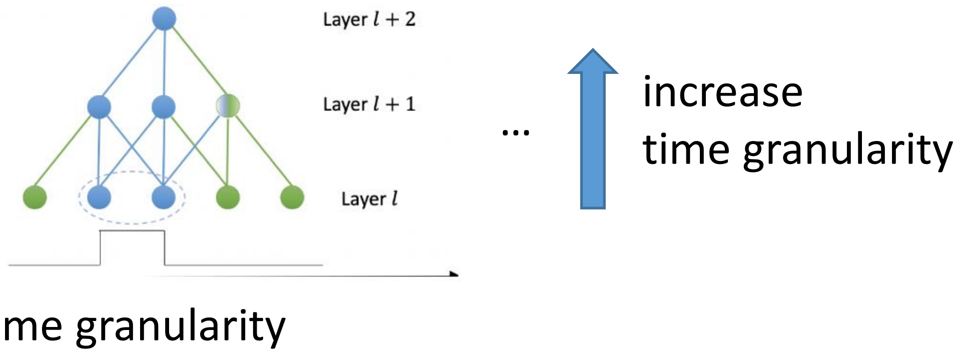
- β holds certain self-explaining power.

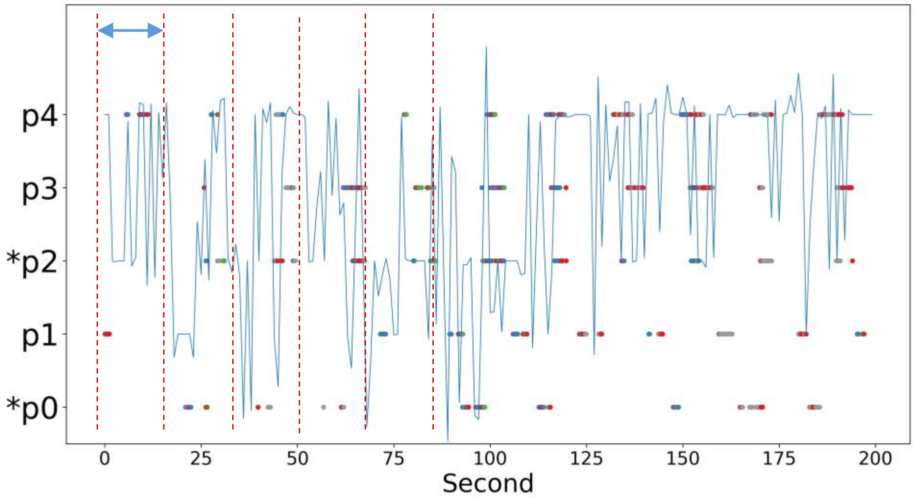
Set-Temporal Convolution --- I

- The order (of interactions) is important at micro level
E.g., A looks at B and then B avoids the eye contact…

- Controllable time granularity --- different micro levels.

Hierarchical temporal convolution network





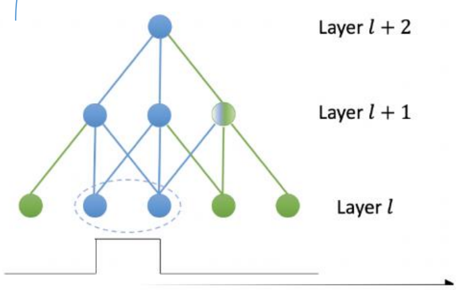
Set-Temporal Convolution --- II

- The order (of interactions) is not important at macro level

E.g., A looks at B and then B avoids the eye contact…
(in the first 100s v.s. last 100s)

- Handling the long time span

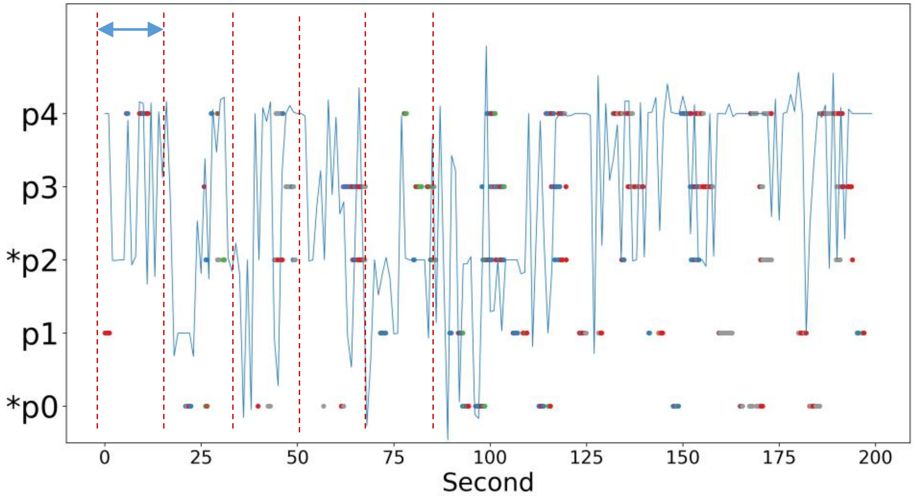
Set – pooling {f1, f2, …, fk} -> f



…

The order of fi, fj does not impact the output

time granularity



Dataset & Task

	Task No. and Task	Dataset	Dynamic Network Sequences	Time Steps (Avg.)	Nodes	Interactions†
Ours	(1) Dominance Identification	RESISTANCE	956	2, 514	4780	4.007×10^6
	(1) Dominance Identification	ELEA	21*	1, 350	84	6.474×10^3
	(2) Deception Detection	RESISTANCE	2, 157	1, 800	10, 785	2.439×10^7
	(3) Nervousness Detection	RESISTANCE	1, 097	1, 800	5, 485	4.910×10^7
	(4) Community Detection	CIAW	1	20	92	2.149×10^4

* We use all the four-person games in ELEA; † We count all the interactions with probability ≥ 0.5.

Table 1: General statistics of the dynamic networks used for representation learning.

Task No. and Task	Description
(1) Dominance Identification	Predict the most dominant person in each interaction event
(2) Deception Detection	Predict *all* the lying people in each interaction event
(3) Nervousness Detection	Predict the most nervous person in each interaction event
(4) Community Detection	Predict each person's community belonging

Performance

RESISTANCE (our dataset):

	Dominance Identification		Deception Detection		Nervousness Detection	
	Method	Performance	Method	Performance	Method	Performance
Baselines	MKL	0.879	FAU	0.608	LR.	0.493
	DELF	0.889	TGCN-L	0.550	RF.	0.678
	GDP-MLP	<u>0.917</u>	LiarOrNot	<u>0.665</u>	GCN-LSTM	0.702
	GDP-RF	0.878	ADD	0.632	Facial Cues	<u>0.733</u>
Ours	-	**0.923** (±0.009)	-	**0.742** (±0.021)	-	**0.769** (±0.023)

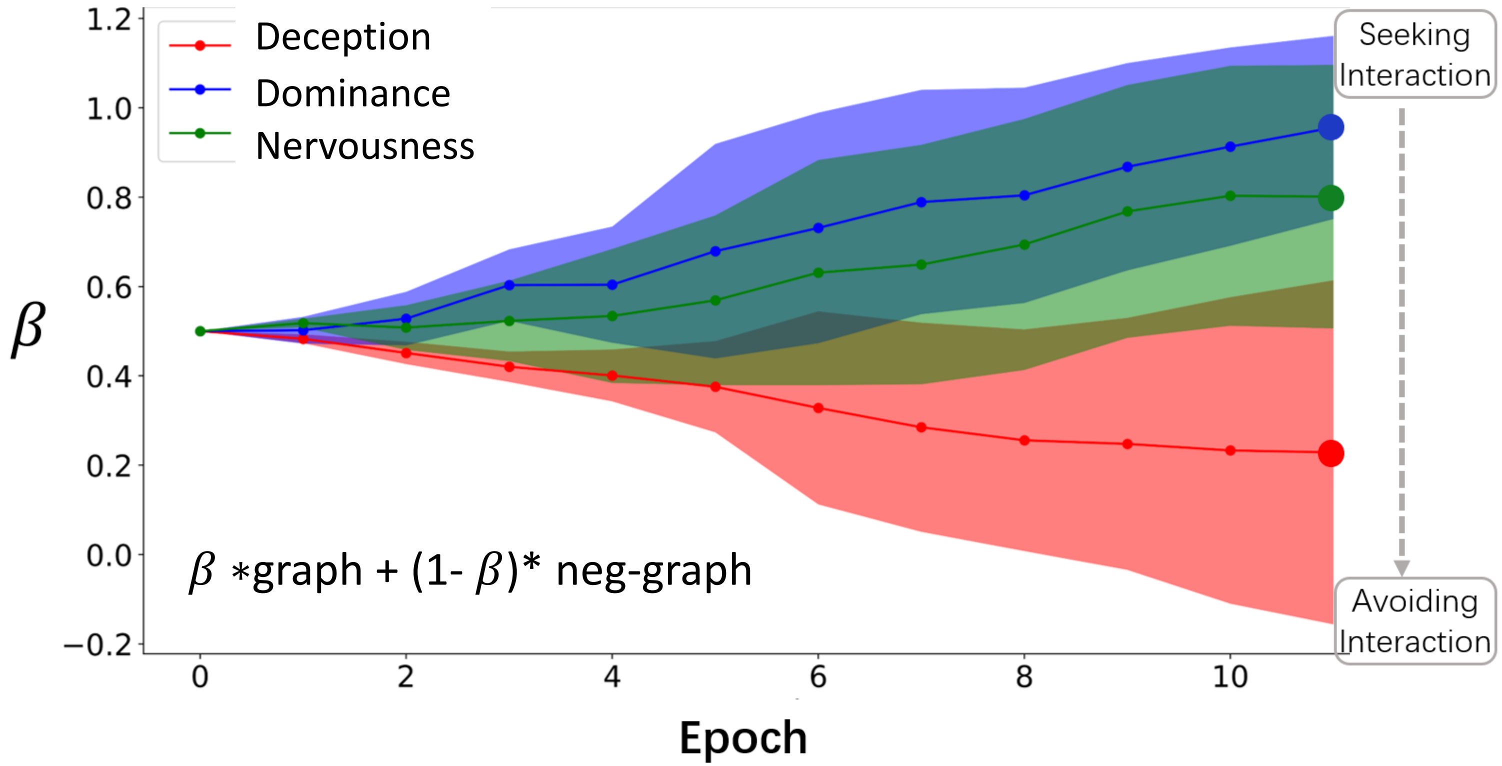
- MKL, DELF, GDP, FAU, ADD, LiarOrNot, Facial Cues --- Feature-analysis-based approaches
- TGCN-L, GCN-LSTM --- Generic approaches for representation learning over dynamic networks

Other datasets (CIAW and ELEA):

Community Detection	
Method	Perform.
WD-GCN	0.813
CD-GCN	0.819
GCN-LSTM [40]	0.601
EvolveGCN	<u>0.893</u>
-	**0.929** (±0.011)

Dominance Identification	
Method	Performance
Aran et al.	0.657
Okada et al.	0.677
DELF-FV	<u>0.765</u>
DELF-Hist	0.745
Humans	0.686
Ours	**0.774**

Self-Explanation #I



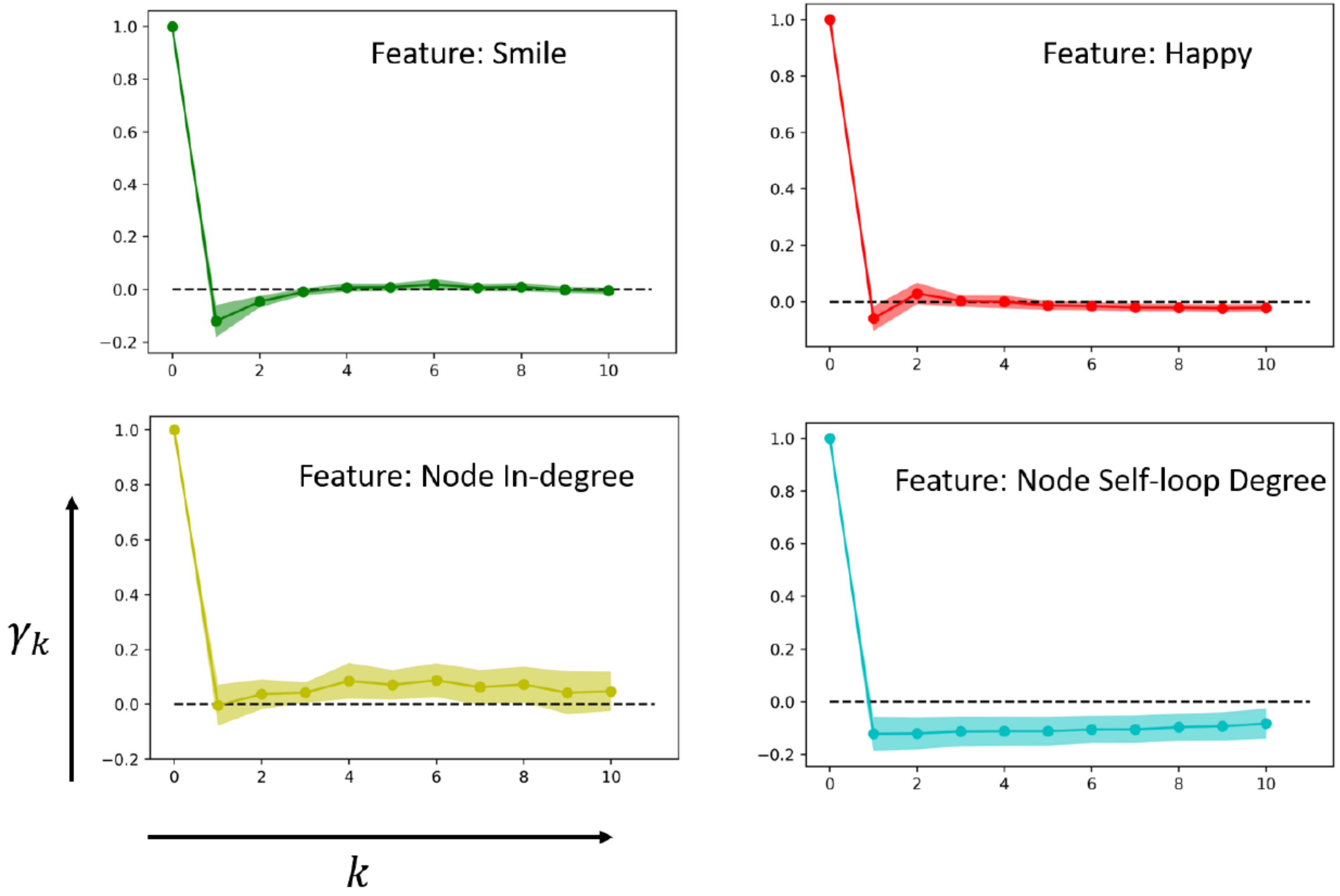
Deception Detection, Dominance Identification, Nervousness Detection run on RESISTANCE (our dataset)

Insights from Graph Weight β:

- **Deception Detection:** Avoiding interaction is more informative
- **Dominance Identification:** Seeking interaction is more informative
- **Nervousness Detection:** A mixture in between

--- Coincided with the findings of psychological theory [K. Rayner 1998]

Self-Explanation #II



Nervousness Detection. γ_k weights for one features at k-hop.

Insights from Diffusion weights γ_k, k: #hops

- **Facial expression features:**

Do not impact others much in long-hop propagation

- **Others' eye focus (In-degree) and Ones' eye focus (self-loop degree)**

Impact much

--- When someone is gazed at by others, he tends to be nervous

Thanks for your attention! Questions

How about other proposed deep models?

- General graph neural networks for dynamic networks (mostly for future-interaction prediction) are hard to be applied to --- deception detection, dominance identification, nervousness detection…

- Goal: Design the architecture of a deep model that is

1. General enough to cover different prediction tasks
2. Suitable for our case (predicting social status)
3. Self-explainable to some extend