AI4media
ARTIFICIAL INTELLIGENCE FOR THE MEDIA AND SOCIETY

AIDA
AI Doctoral Academy initiative

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 951911

1

!

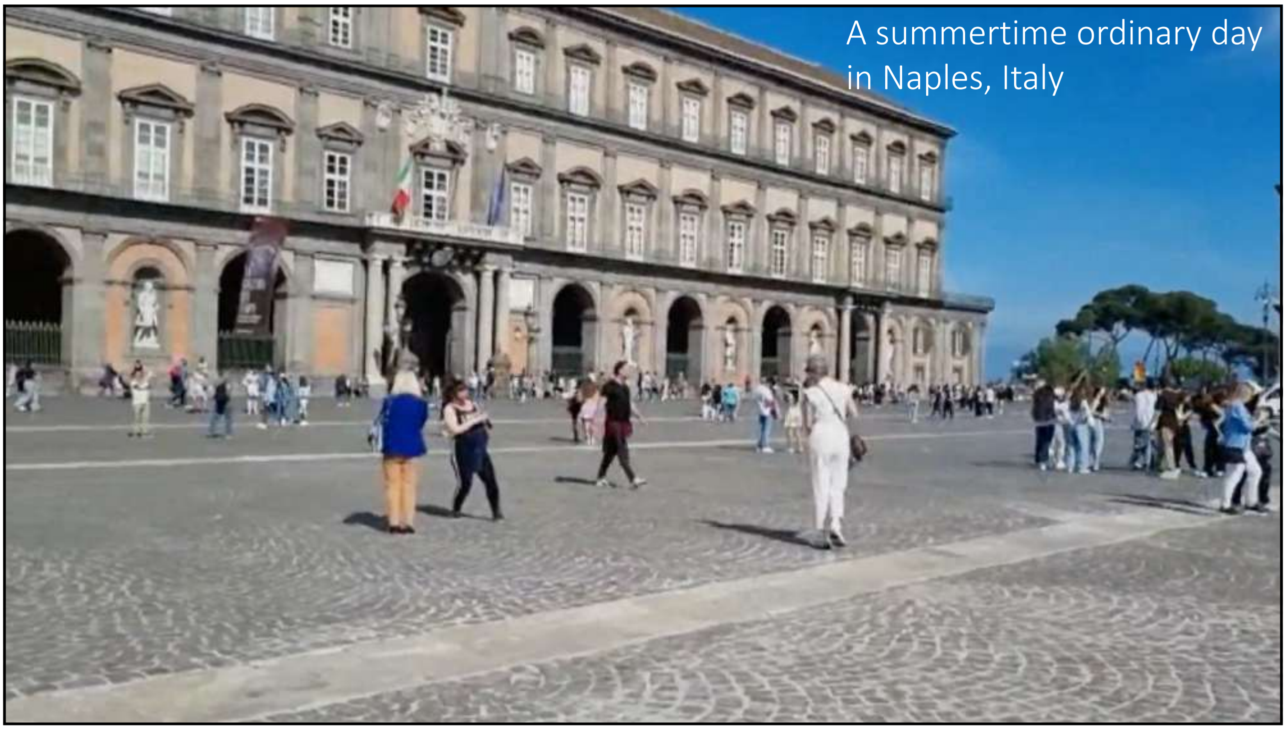
AI Excellence Lecture Series

Social interaction in trajectory prediction with Memory Augmented Networks

July 5, 2022

Prof. Alberto del Bimbo
Università di Firenze, Italy

"



#

Predicting human social behavior

Pedestrians move with complex and stochastic behavior
Usually follow common sense and specific social rules
Often walk in groups

Observe near people's behavior anticipating
what will happen in the neighbourhood……

We aim at emulating human forecasting
i.e. predicting human trajectories by
modelling social interactions

$

Why so important

Essential for autonomous moving platforms like self-driving cars
or social robots that will share the same ecosystem as humans
or surveillance systems where helping identifying suspicious activities….

Accurate prediction of the future motion of other moving agents in their working space
is essential for the following safety decision-making and control processes,
giving mobility to handicapped people…..

%

Problem definition

Past
Future

All trajectories are sequences of top-view 2D spatial coordinates in a fixed reference frame independent from the agents

Given a social context $S = \{!^{!}, i = 0, ..., N - 1\}$ defined as the set of trajectories representing N moving agents, the task of trajectory prediction is the problem of predicting the future positions of each agent, given their past positions

Trajectory prediction is an inherently multimodal problem
Multiple outcomes are possible

Modeling both temporal and spatial reasoning is needed

&

Literature: Recurrent Network and Pooling-based

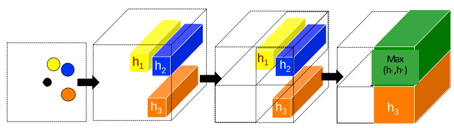
Social GAN: Socially Acceptable Trajectories with Generative Adversarial Networks
A. Gupta et al. , *Proc. of CVPR* 2018

Temporal features first
Spatial features afterwards
LSTM Memory network

- *LSTM recurrent network* to encode the motion of each agent
- *Pooling* to aggregate information of individuals
- *LSTM recurrent network* to decode and generate the future trajectory conditioned on person state and pooled vector

$$L_{variety} = \min_k \|Y_i - \hat{Y}_i^{(k)}\|_2,$$

Trained using GAN
- a generative model that captures the data distribution
- a discriminative model that estimates the probability that a sample came from the training data



Pooling computes relative positions between the red and all other people: the positions are concatenated with each person’s hidden state and processed independently

Passing the input coordinates through a MLP followed by Max-Pooling

Recurrent architectures are parameter inefficient and expensive in training. Temporal ordering is lost and agent-wise knowledge is disregarded in favor of a coarse global descriptor

The pooling aggregation in feature states is not intuitive in modelling interactions between people as the physical meaning of feature states is difficult to interpret

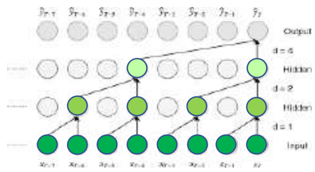
GAN generates highly diverse trajectories but tend to neglect the physical structure of the environment. The resulting trajectories are not necessarily feasible, and often do not fully cover multiple possible directions that a pedestrian can take

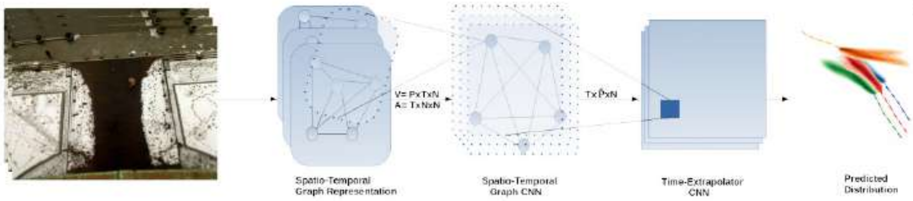
Literature: Graph-based and Attention pooling

Social-STGCNN: A Social Spatio-Temporal Graph Convolutional Neural Network for Human Trajectory Prediction
A. Mohamed et al. , *Proc. CVPR* 2020

Spatial features first
Temporal features afterwards
Stateless system

- *Spatial graph* represents the relative locations v^i_t of the agents in a scene at time t
- *Spatio-temporal graph* represent T frames
- *Spatio-temporal Graph Convolution Neural Network* creates a spatio-temporal embedding
- *Time-Extrapolator-CNN* predicts future trajectories through convolution operators





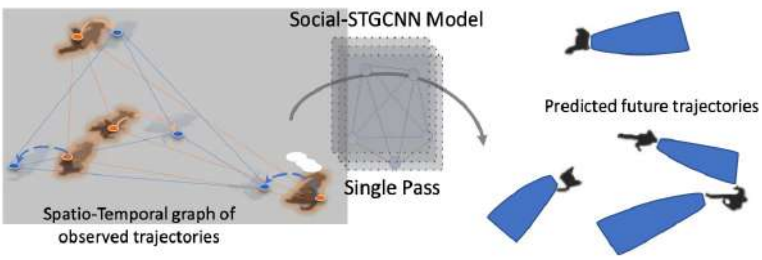
The topology of $G_1, \ldots, G_T$ is the same, while different attributes are assigned to v^i_t when t varies

)

Graphs are a natural more direct, intuitive and efficient way to model pedestrians interactions than aggregation based methods

The Convolution operation over graphs is a weighted aggregation of target node attributes with the attributes of its neighbor nodes
Attention-based pooling: attention weights according to euclidean distance

A stateless system
Single pass prediction: high gain in parameter efficiency wrt LSTM



Models relationships between signals but relies on some fixed-size hidden representation blending them together

Only models the interactions of spatially proximal trac-agents
Ignore the influence beyond the given spatial limits

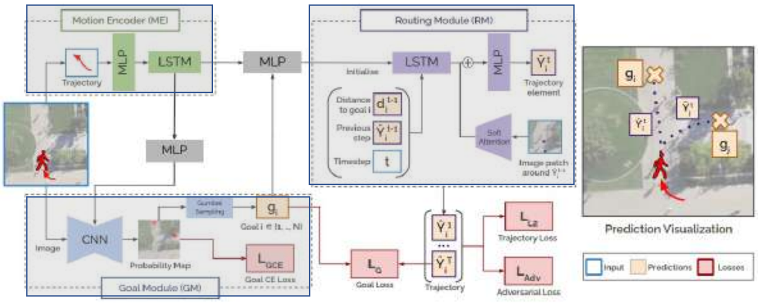
! *

Literature: Recurrent Network and Intention estimation-based

Goal-GAN: Multimodal Trajectory Prediction Based on Goal Position Estimation
P. Dendorfer et al. *Proc. of ACCV* 2020

Temporal features first
Spatial features afterwards
LSTM Memory network

- *Motion Encoder*: LSTM encodes the speed and direction of motion of the agent's past trajectory
- *Goal Module*: predicts the most likely target positions of the agent combining visual scene information and agent's dynamics
- *Routing Module*: estimates a set of plausible trajectories that route towards the estimated goal



$$\mathcal{L} = \lambda_{Adv}\,\mathcal{L}_{Adv} + \mathcal{L}_{L2} + \lambda_G\,\mathcal{L}_G + \lambda_{GCE}\,\mathcal{L}_{GCE}.$$
$$\mathcal{L}_{Adv} = \frac{1}{2}\mathbb{E}\left[(D\,(X,Y)-1)^2\right] + \frac{1}{2}\mathbb{E}\left[D(X,\hat{Y})^2\right]$$
$$\mathcal{L}_{L2} = \min_k \|Y - \hat{Y}^{(k)}\|_2$$
$$\mathcal{L}_G = \|g - \hat{Y}^{t_{pred}}\|_2.$$
$$\mathcal{L}_{GCE} = -\log\,(p_i)$$

GAN to train the trajectory generator to output realistic and physically feasible trajectories

! !

Two-stages process
First estimates a posterior over possible goals taking into account the dynamics of the pedestrian and the visual scene context
Then predicts trajectories that route towards these estimated goals

Realistic multimodal: the *Goal module* estimates a multi-modal probability distribution over the possible goal positions which is used to sample a potential goal during the inference

ETH | Zara 2 | Hyang 4 0 | Coupa 1

Solely estimating trajectory goals neglects the social context so affecting agent trajectories after the present time-step

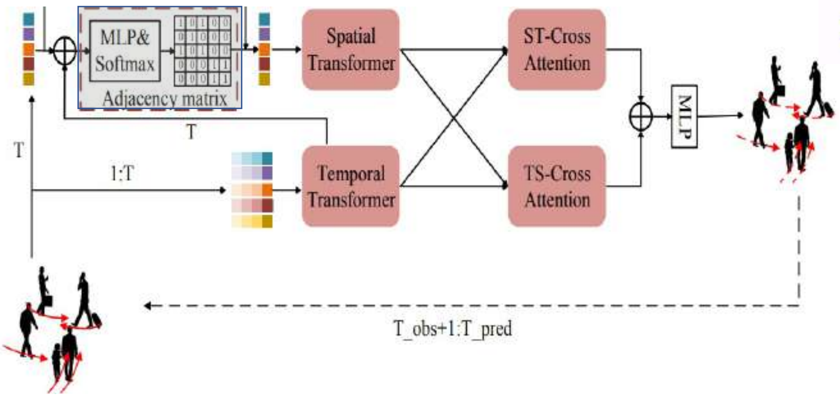
! "

Literature: Transformer attention-based

GA-STT: Human Trajectory Prediction with Group Aware Spatial-Temporal Transformer
L. Zhou et al., IEEE Robotics and Automation Letters, 2022

Temporal features first
Spatial features afterwards
Transformer attention

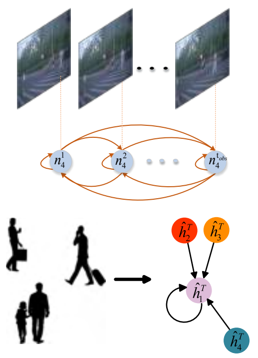
- *Adjacency matrix* is learned and used to enhance the individual representation with group constraints
- *Spatial Transformer* and *Temporal Transformer* are respectively used to extract social interaction and temporal features
- *Cross-attention modules* capture the spatial-temporal dependencies



!#

The *Temporal Transformer* takes the temporal features of pedestrians i with observation from 1 to T as input and outputs an enhanced feature with temporal dependencies

The *Spatial Transformer* treats the crowd at time T as a graph to capture the spatial interaction



$$Q^i = f_Q\left(\left\{\overline{h}_i^t\right\}_{t=1}^T\right)$$
$$K^i = f_K\left(\left\{\overline{h}_i^t\right\}_{t=1}^T\right)$$
$$V^i = f_V\left(\left\{\overline{h}_i^t\right\}_{t=1}^T\right)$$
$$\mathrm{Att}\left(Q^i, K^i, V^i\right) = \frac{\mathrm{Softmax}\left(Q^i K^{i\mathrm{T}}\right)}{\sqrt{D}} V^i$$

$$Q^T = f_Q\left(\left\{\hat{h}_i^T\right\}_{i=1}^N\right)$$
$$K^T = f_K\left(\left\{\hat{h}_i^T\right\}_{i=1}^N\right)$$
$$V^T = f_V\left(\left\{\hat{h}_i^T\right\}_{i=1}^N\right)$$
$$\mathrm{Att}\left(Q^T, K^T, V^T\right) = \frac{\mathrm{Softmax}\left(Q^T K^{T\mathrm{T}}\right)}{\sqrt{D}} V^{\mathrm{T}}$$

Cross attention modules capture spatial and temporal dependencies and consider them integrally
- For the *Spatial-temporal cross attention* the spatial feature vector is the query and the individual temporal feature is key and value
- For the *Temporal-spatial cross attention,* the individual temporal feature vector is the query the spatial feature is key and value

The output of these two cross attention are fused by a fully connected layer to generate the enhanced individual representation

!$

Strengths and weaknesses of Transformer-based

More powerful than LSTM in modelling temporal dependencies due to self-attention mechanism

More powerful network structure than social pooling for spatial interaction modelling

The feedforward nature of Transformers makes them efficient on modern hardware

Limits:
Inability to track very long sequences and process hierarchical inputs or algorithmic tasks

Only a fixed number of transformations can be applied to its internal states
The total number of transformations between the input and output is limited by the sub-layers depth

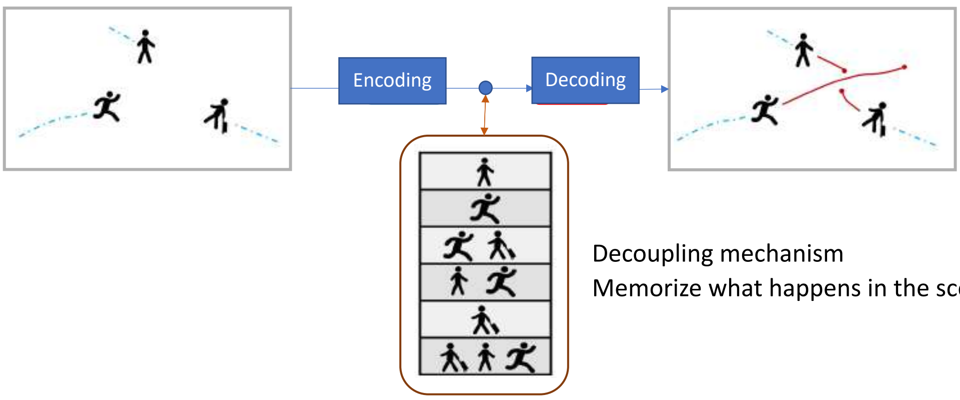
At each layer, the representations for the input sequence are treated in parallel. The high-level representations from the past are not exploited to compute the current representation

Both essential and non-essential information are considered, with more and less attention

!%

An alternative option: usage of a working memory

Storage of items of relevant information to decision-making
Avoids blending past information into a single latent state, but instead uses memory to keep track of relevant cues across time and store them separately to be recalled



Decoupling mechanism
Memorize what happens in the scene step by step

Short-term
episodic memory

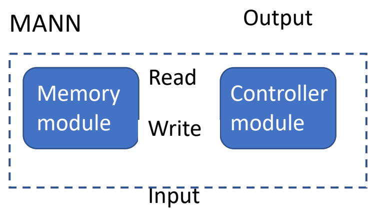
!&

Memory Augmented Neural Networks

A Recurrent Network-based Memory Controller and an external trainable Memory
The Memory Controller is trained to write all the examples in memory and to read what is necessary to produce the output

Keeps in memory a set of independent states instead of incrementally creating a state
This helps to find the structure in the training data and to generalize to sequences in algorithmic tasks

Think the Controller network as the CPU and the external memory as the RAM



! '

Memory Augmented Neural Network model

Memory Network

Memory Augmented Network

Inputs are fed to LSTM one-by-one
LSTM has only one chance to look at an input symbol

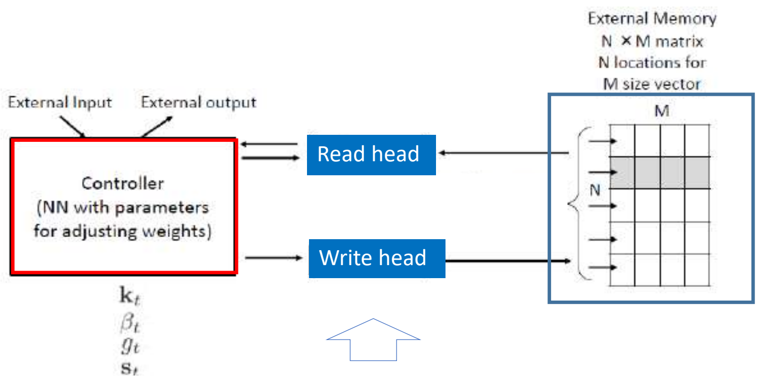
Place all input symbols in memory and
let the model decide which part it reads next

! (

Neural Turing Machine
Graves, G. Wayne, I. Danihelka, ArXiv preprint, 2014

Neural Turing Machine is a MANN that learns to read and write data from/into the external memory at different time steps to solve a given task

- *Network Controller*: the interface between the input and and the memory through read and write heads
- *Memory Bank*: an array of vectors



The heads, with the feature generated by Controller, compute addressing to read/write in memory

Credits S. Malekmohammadi

!)

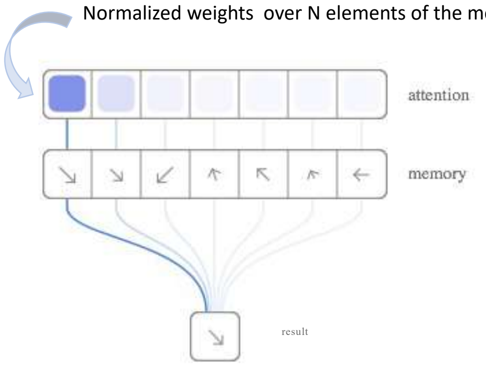
NTM blurry operations

The operations *argmax* o *select index* are not differentiable: use blurry operations i.e. interact to a greater or lesser degree with all the elements in memory rather than addressing a single or few element directly

The degree of blurriness is determined by an attentional focus mechanism that constrains each Read and Write operation to interact with a small portion of the memory while ignoring the rest

Normalized weights over N elements of the memory

$$\sum_i w_i = 1$$



A weight of 1 focuses all the attention on the corresponding memory location
A weight of 0 discards that memory location

Credits A. Graves et al

" *

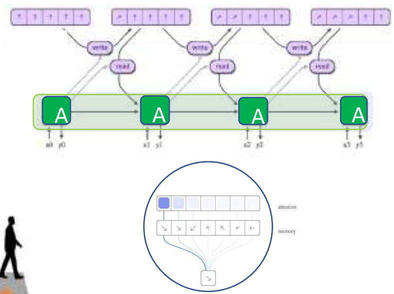
Memory networks compared

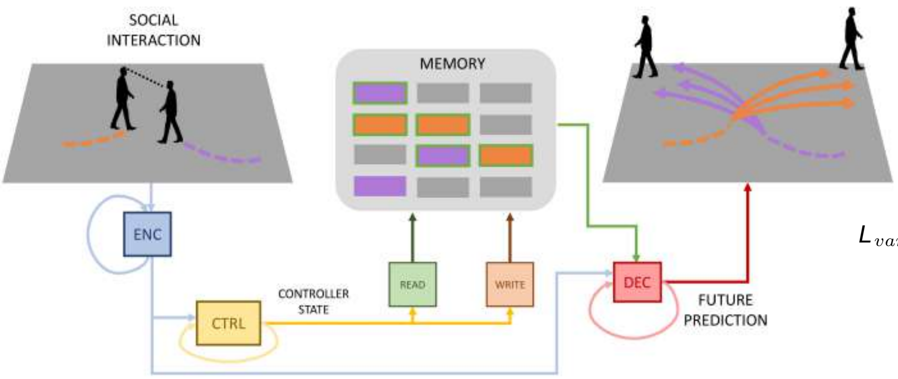
RNN, LSTM, GRU	Memory Augmented Neural Networks
Memory is a single hidden state vector that encodes all the temporal information	Add an external memory matrix with increased storage capacity
Memory is addesssable as a whole All the past information is encoded in the state vector	Memory is element-wise addressable. Rely on attention to work
State to state transition is unstructured and global	State to state transitions are obtained through read/write operations
Find some structure in the training data	Find the structure in the training data, but also generalize to long sequences in algorithmic tasks
The number of parameters is tied to the size of the hidden state	The number of parameters is not tied to the size of the memory.
	Ability to track long sequences and process hierarchical inputs, maintaining an internal state for long time

" !

SMEMO: Social Memory for Trajectory Forecasting
F. Marchetti, F. Becattini, L. Seidenari, A. Del Bimbo, ArXiv preprint, 2022

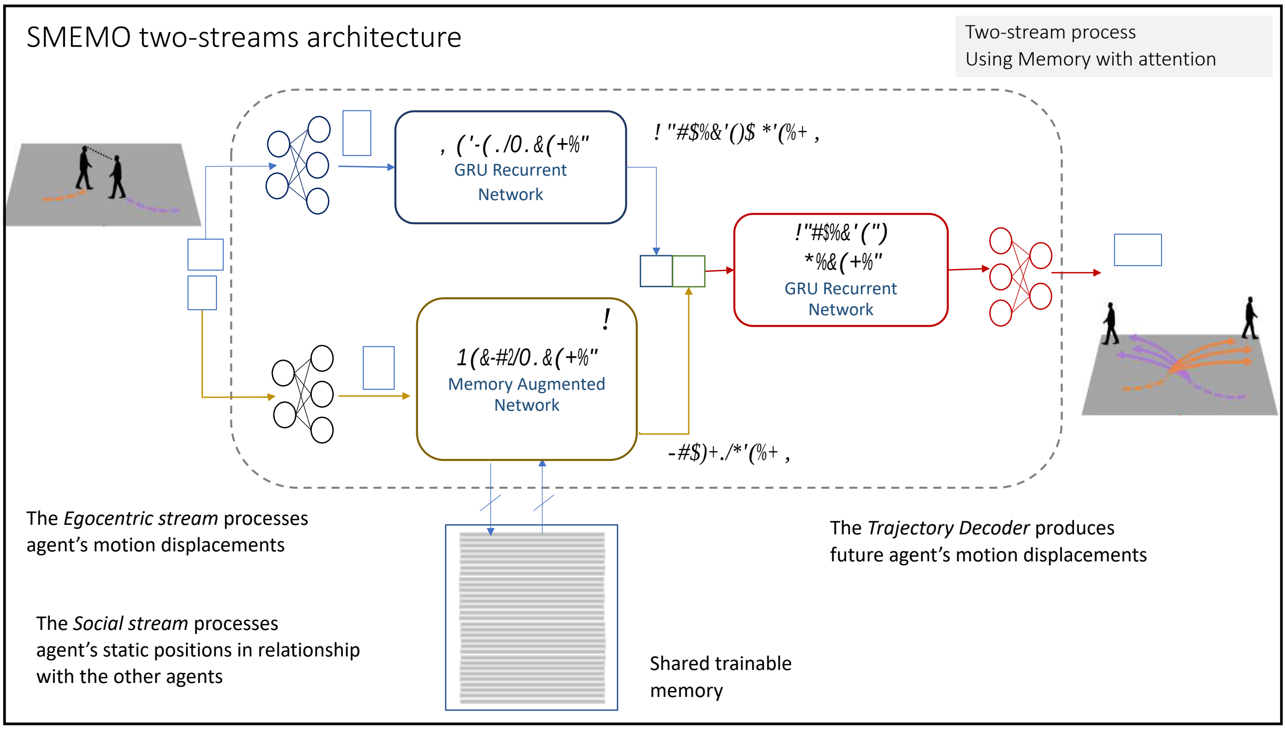
End-to-end differentiable model
Based on Memory Augmented Neural Network



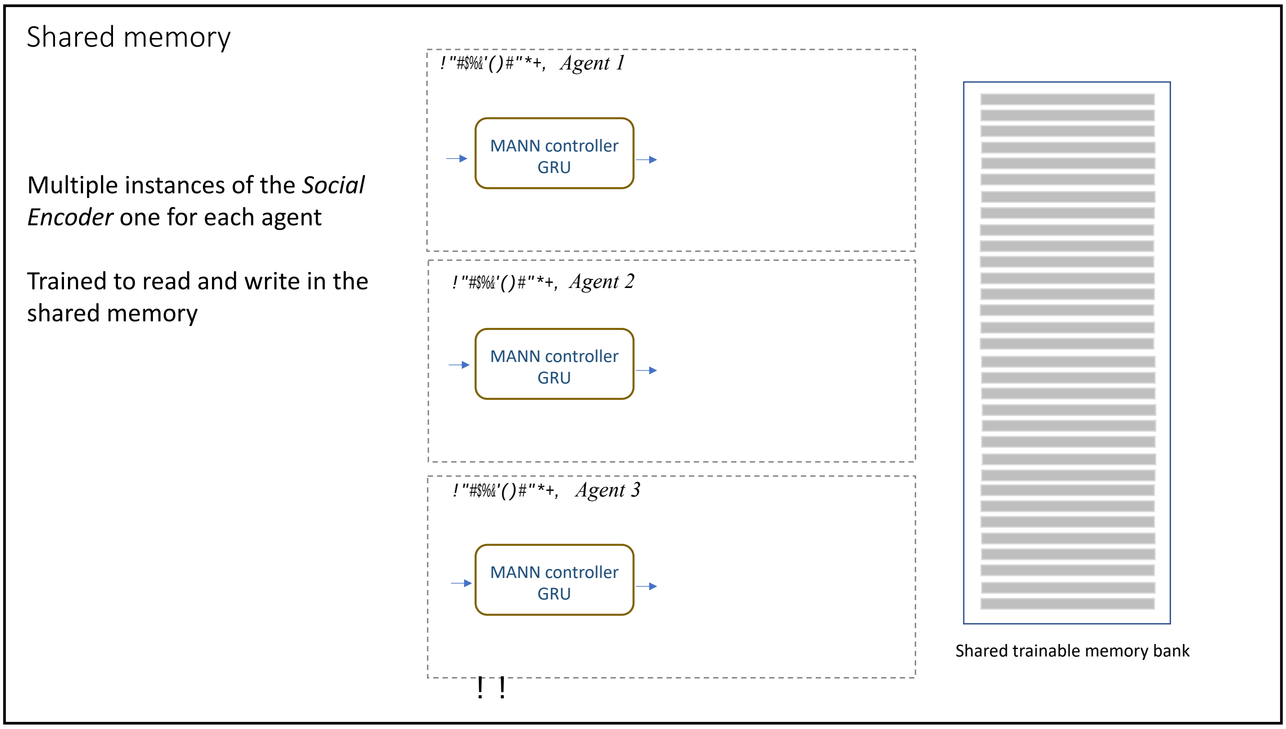


$$L_{variety} = \min_k \|Y_i - \hat{Y}_i^{(k)}\|_2,$$

""

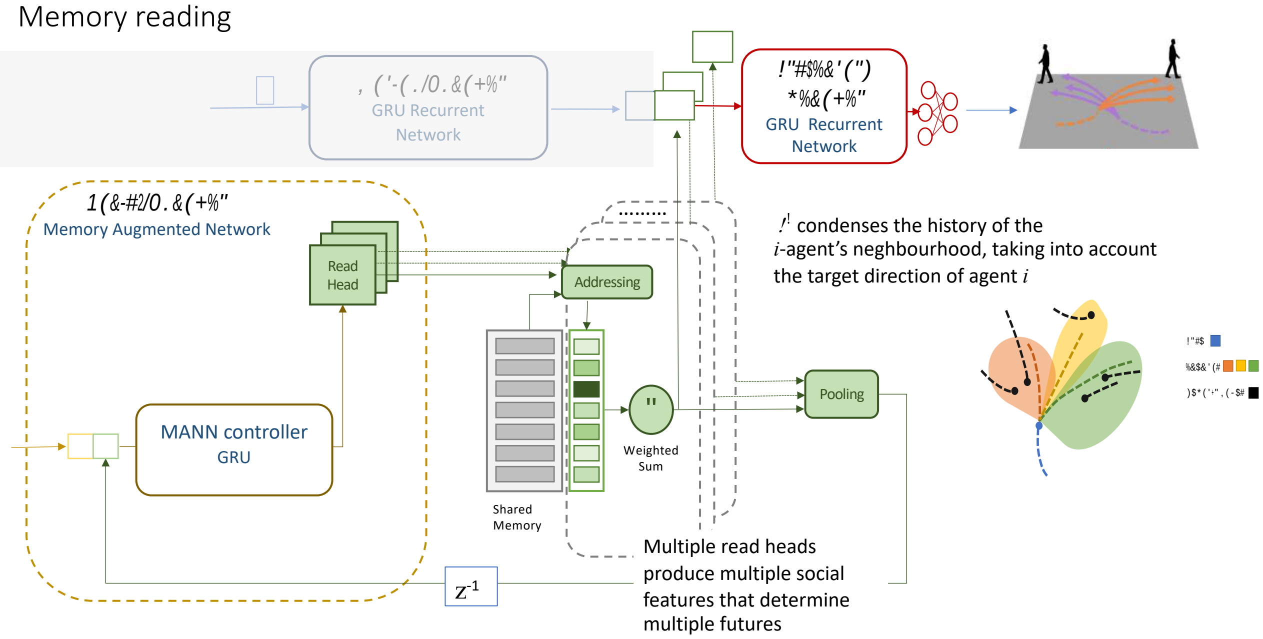

SMEMO two-streams architecture
Two-stream process
Using Memory with attention
GRU Recurrent
Network
GRU Recurrent
Network
Memory Augmented
Network
The *Egocentric stream* processes
agent's motion displacements
The *Social stream* processes
agent's static positions in relationship
with the other agents
Shared trainable
memory
The *Trajectory Decoder* produces
future agent's motion displacements

"#


Shared memory
Multiple instances of the *Social Encoder* one for each agent
Trained to read and write in the shared memory
Agent 1
MANN controller
GRU
Agent 2
MANN controller
GRU
Agent 3
MANN controller
GRU
Shared trainable memory bank

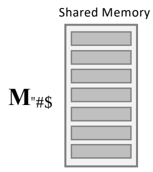
"$

Memory reading

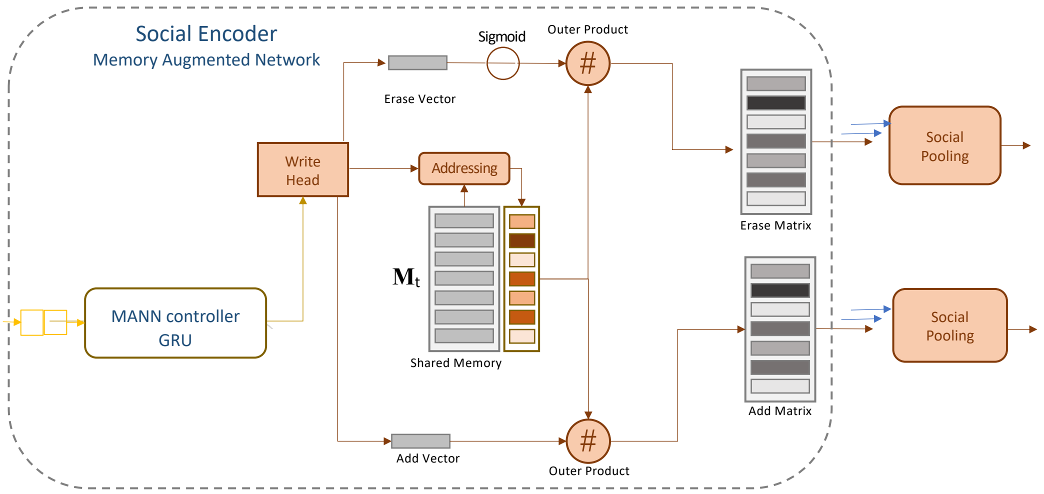


"%

Memory writing



The memory controller updates memory for the next timesteps
Erase and Add Matrix are generated using the addressed memory content
Social Pooling guarantees invariance to agent ordering



"&

Experiment example 1

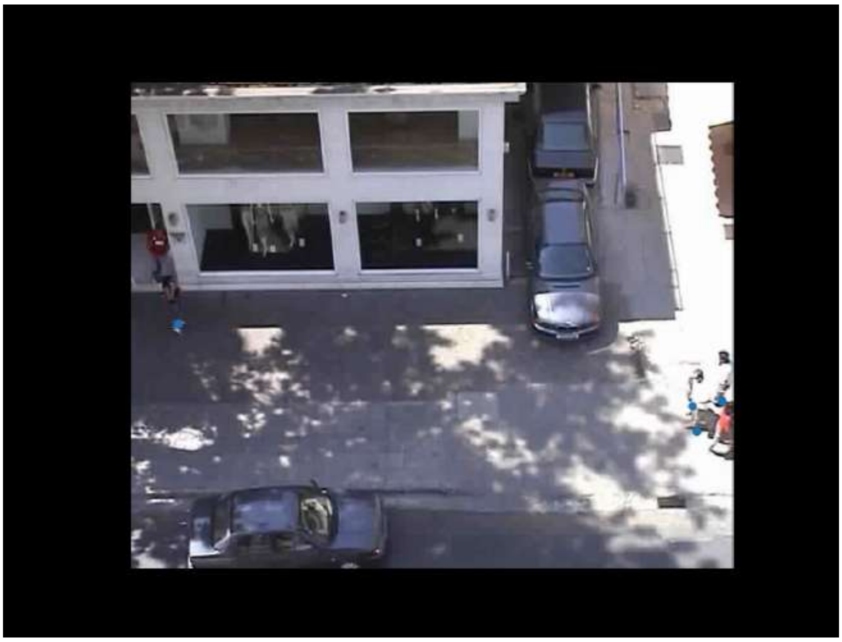
ETH/UCY
Univ

Past trajectory in blue
Predictions (3) in red

" '

Experiment example 2

ETH/UCY
Zara



Past trajectory in blue
Predictions (3) in red

"(

Examples with Ground truth

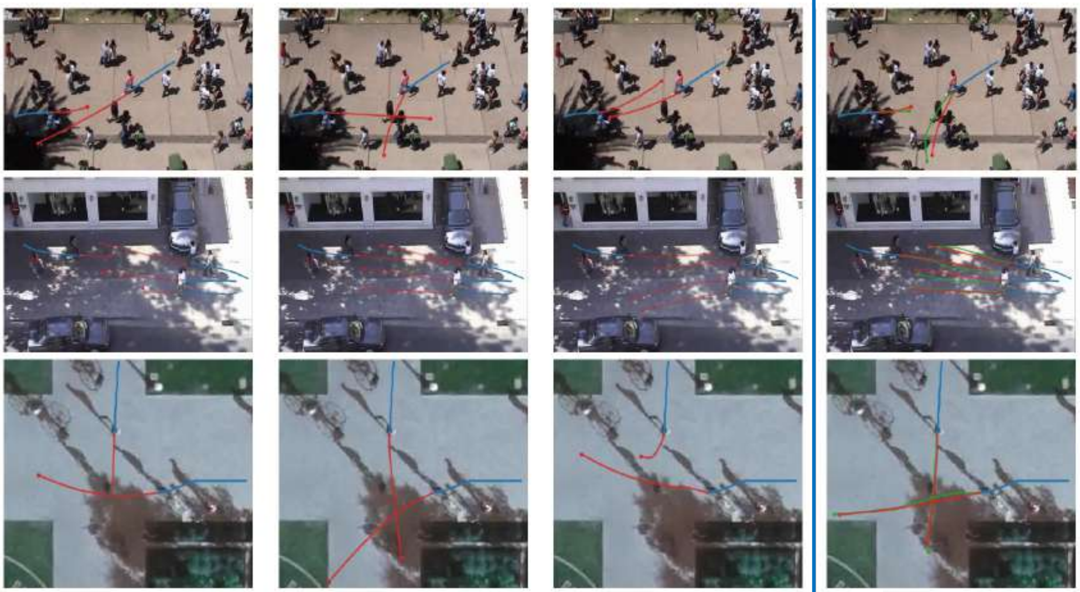
Past trajectory in blue
Predictions (3) in red

Ground truth in green

ETH/UCY Univ

ETH/UCY Zara

SDD



Social Multimodal Prediction

GT + best prediction

")

Comparative Tables - ETH/UCY, SDD datasets

K: number of predictions

ETH/UCY

Method (K=20)	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVERAGE
Social-GAN	0.81/1.52	0.72/1.61	0.60/1.26	0.34/0.69	0.42/0.84	0.58/1.18
SoPhie	0.70/1.43	0.76/1.67	0.54/1.24	0.30/0.63	0.38/0.78	0.54/1.15
CGNS	0.62/1.40	0.70/0.93	0.48/1.22	0.32/0.59	0.35/0.71	0.49/0.97
S-BiGAT	0.69/1.29	0.49/1.01	0.55/1.32	0.30/0.62	0.36/0.75	0.48/1.00
MATF	1.01/1.75	0.43/0.80	0.44/0.91	0.26/0.45	0.26/0.57	0.48/0.90
GOAL-GAN	0.59/1.18	0.19/0.35	0.60/1.19	0.43/0.87	0.32/0.65	0.43/0.85
Transformer	0.61/1.12	0.18/0.30	0.35/0.65	0.22/0.38	0.17/0.32	0.31/0.55
PECNet	0.54/0.87	0.18/0.24	0.35/0.60	0.22/0.39	0.17/0.30	0.29/0.48
Trajectron++	**0.39**/0.83	**0.12/0.19**	**0.22**/0.43	**0.17/0.32**	**0.12/0.25**	**0.20**/0.40
SMEMO	0.45/**0.67**	0.15/0.22	0.23/**0.41**	0.19/0.33	0.15/0.26	0.23/**0.37**

SDD

K=5			K=20					
Method	ADE	FDE	Method	ADE	FDE	Method	ADE	FDE
DESIRE	19.25	34.05	Social-GAN	27.25	41.44	EvolveGraph	13.90	22.90
Ridel et al.	14.92	27.97	Trajectron++	19.30	32.70	Goal-GAN	12.20	22.10
PECNet	12.79	25.98	SoPhie	16.27	29.38	SimAug	10.27	19.71
TNT	12.23	21.16	CF-VAE	12.60	22.30	PECNet	9.96	15.88
SMEMO	**11.64**	**21.12**	P2TIRL	12.58	22.07	**SMEMO**	**8.11**	**13.06**

#*

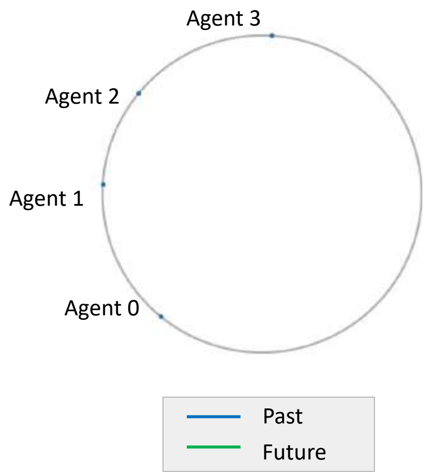
Synthetic Social Agents dataset - SSA

From 3 to 10 agents
starting from a point on a circumference…
- going towards the center with constant speed
- different agent speeds: 8–12 frame/sec

When two agents get close:
- the agent with greater speed passes
- the other stops

Trajectories:
- Past: 3,2 sec
- Future: 4,8 sec

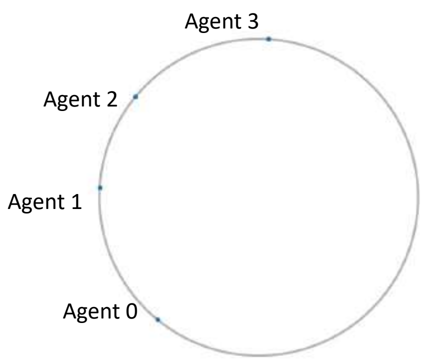
Suited for learning algorithmic tasks in trajectory prediction



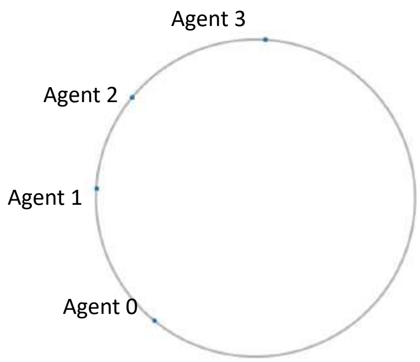
#!

Learning in simple scenarios – SSA dataset

With 4 agents



GROUND-TRUTH



SMEMO

#"

MANN versus GRU

Managing to slow down predictions when social interactions occur

Agent speeds

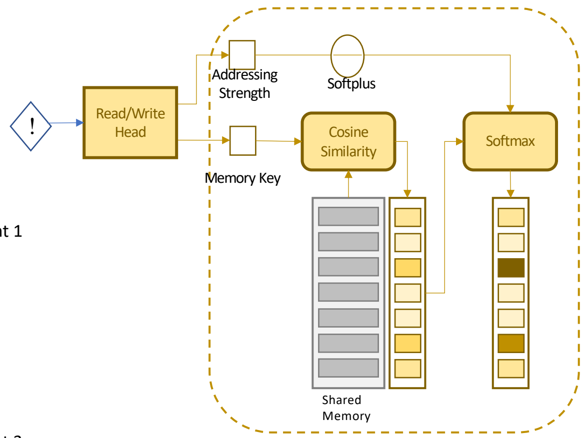
Agent 0

Agent 1

Agent 2

Agent 3

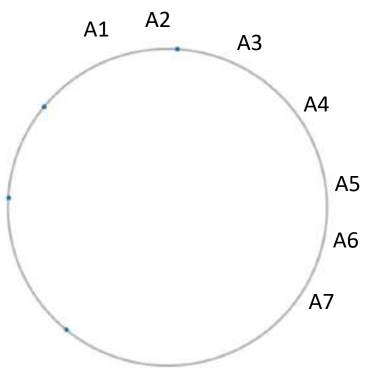
Addressing with attention



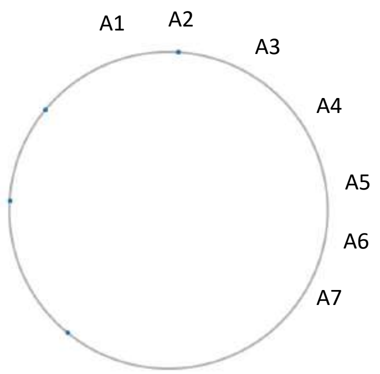
##

Scaling to more crowdy scenarios - SSA dataset

With 7 agents



GROUND-TRUTH



SMEMO

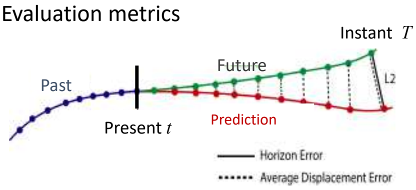
#$

Comparative table – SSA dataset

Method	ADE ↓	FDE ↓	Kendall ↑
Linear	0.091	0.141	0.67
MLP	0.087	0.138	0.65
GRU ENC-DEC	0.087	0.138	0.64
Expert-Goals	0.095	0.149	0.49
PECNet	0.045	0.136	0.71
Trajectron++	0.084	0.132	0.59
Social-GAN	0.051	0.085	0.67
AgentFormer	0.040	0.064	0.70
SMEMO	**0.027**	**0.038**	**0.83**

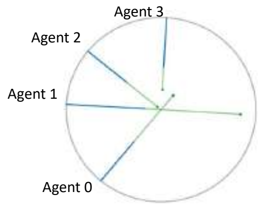
Ablation study	ADE ↓	FDE ↓	Kendall ↑
SMEMO	**0.027**	**0.038**	**0.83**
Memory reset	0.030	0.045	0.79
Zero reading	0.087	0.137	0.64
Random reading	0.087	0.137	0.65
State pooling	0.045	0.069	0.69

Evaluation metrics



FDE Final Displacement Error
ADE Average Displacement Error

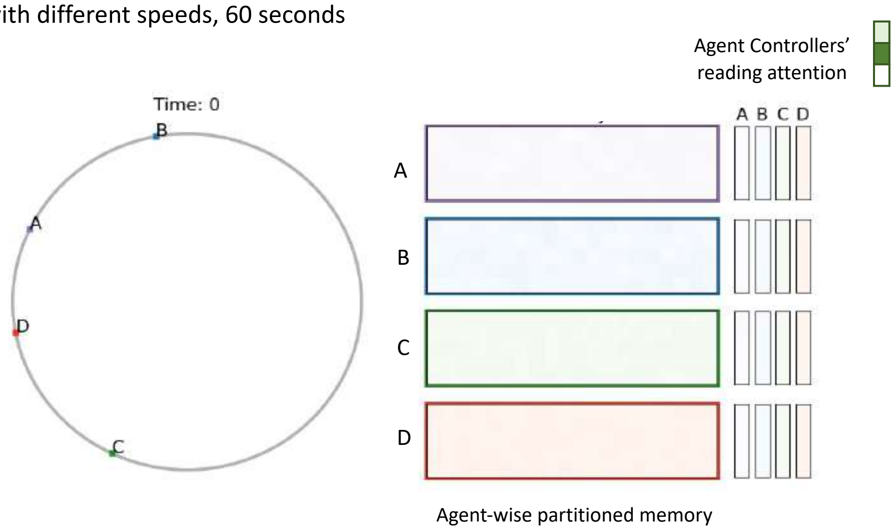
Kendall: agents' order passing through the center



#%

Explainability

Memory-based future forecasting provides explainability
Explains which information is relevant to the the task
4 agents with different speeds, 60 seconds



#&

From 0 to 31

A

Agent B gives some attention to Agent A

B

Agent D focuses on Agent C

C

Agent A focuses on Agent D
Agent C focuses on Agent D

D

Agent C is slightly faster than the other agents
Agent D is faster than its closest agent A and faster than B
Agent A is slightly faster than its closest agent B but
had to stop so It is late wrt B

#'

From 31 to 42

A

Agent B gives some focus on Agent A

B

Agent A, C, D read empty space almost everywhere
so don't stop

C

D

Agent C and D already passsed through the center so
no collision can occur
A is faster than its closest Agent B although being late

#(

From 31 to 42

A

B

C

D

All Agents read empty space

All Agents safely passed through the center

#)

Conclusions

Exploits an episodic working memory to manipulate observations and reason about social interactions

Multimodality: each read controller can generate a different prediction trajectory.
Real trajectories are enforced by the variety loss

Memory: maintains updated internal state indefinitely

Information filtering: model parameters are trained from relevant instances only, avoiding irrelevant information that can disturbe the performance

Explainability: by checking which information items in memory the reading controller focuses on we can understand the task-relevant information. Suitable for safety-sensitive applications

$*

A few references

Other methods

A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2255–2264.

A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese, "Sophie: An attentive gan for predicting paths compliant to social and physical constraints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1349–1358.

F. Giuliari, I. Hasan, M. Cristani, and F. Galasso, "Transformer networks for trajectory forecasting," *arXiv preprint arXiv:2003.08111*, 2020.

N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. Torr, and M. Chandraker, "Desire: Distant future prediction in dynamic scenes with interacting agents," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 336–345.

Y. Yuan, X. Weng, Y. Ou, and K. Kitani, "Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting," *arXiv preprint arXiv:2103.14023*, 2021.

T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Multi-agent generative trajectory forecasting with heterogeneous data for control," *arXiv preprint arXiv:2001.03093*, 2020.

H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Chai, C. Schmid, C. Li, and D. Anguelov, "Tnt: Target-driven trajectory prediction," *ArXiv*, vol. abs/2008.08294, 2020.

P. Dendorfer, A. Osep, and L. Leal-Taixe, "Goal-gan: Multimodal trajectory prediction based on goal position estimation," in *Proceedings of the Asian Conference on Computer Vision (ACCV)*, November 2020.

K. Mangalam, H. Girase, S. Agarwal, K.-H. Lee, E. Adeli, J. Malik, and A. Gaidon, "It is not the journey but the destination: Endpoint conditioned trajectory prediction," *arXiv preprint arXiv:2004.02025*, 2020.

Z. He and R. P. Wildes, "Where are you heading? dynamic trajectory prediction with expert goal examples," in *Proceedings of the International Conference on Computer Vision (ICCV)*, Oct. 2021.

V. Kosaraju, A. Sadeghian, R. Martin-Martin, I. Reid, H. Rezatofighi, and S. Savarese, "Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks," in *Advances in Neural Information Processing Systems*, vol. 32. Curran Associates, Inc., 2019.

D. Ridel, N. Deo, D. Wolf, and M. Trivedi, "Scene compliant trajector forecast with agent-centric spatio-temporal grids," *IEEE Robotics an Automation Letters*, vol. 5, no. 2, pp. 2816–2823, 2020.

J. Li, H. Ma, and M. Tomizuka, "Conditional generative neural system for probabilistic trajectory prediction," 11 2019, pp. 6150–6156.

T. Zhao, Y. Xu, M. Monfort, W. Choi, C. Baker, Y. Zhao, Y. Wang, and Y. N. Wu, "Multi-agent tensor fusion for contextual trajectory prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 12 126–12 134.

With MANN

A. Graves, G. Wayne, and I. Danihelka, "Neural turing machines," *arXiv preprint arXiv:1410.5401*, 2014.

A. Graves, G. Wayne, M. Reynolds, T. Harley, I. Danihelka, A. Grabska-Barwińska, S. G. Colmenarejo, E. Grefenstette, T. Ramalho, J. Agapiou *et al.*, "Hybrid computing using a neural network with dynamic external memory," *Nature*, vol. 538, no. 7626, pp. 471–476, 2016.

F. Marchetti, F. Becattini, L. Seidenari, and A. Del Bimbo, "Mantra: Memory augmented networks for multiple trajectory prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.

C.Xu,, W. Mao, W. Zhang, S. Chen , ''Remember Intentions: Retrospective-Memory-based Trajectory Prediction'' , in *Proceedings of the International Conference on Computer Vision and Pattern Recognition* (CVPR), June 2022

and many others…

$!

Thank you for your attention

Prof. Alberto Del Bimbo
Università degli Studi di Firenze
Dept. Ingegneria dell'Informazione
Email alberto.delbimbo@unifi.it

$"