Chasing Ghosts : Instruction Following as Bayesian State Tracking

Peter Anderson*[1], Ayush Shrivastava*[1], Devi Parikh[1,2], Dhruv Batra[1,2], Stefan Lee[3]

Georgia Institute of Technology[1] Facebook AI Research[2] Oregon State University[3] * denotes equal contribution

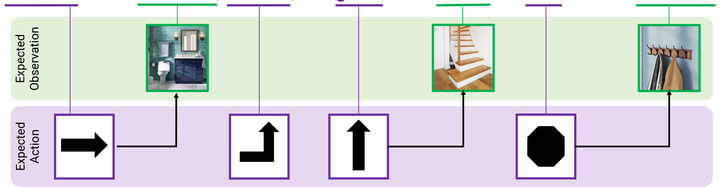
arXiv | Video | Code

NEURAL INFORMATION PROCESSING SYSTEMS

1 Intuition: Unpacking a Navigation Instruction

A **visually-grounded navigation instruction** can be interpreted as a sequence of expected observations and actions an agent following the correct trajectory would encounter and perform.

Walk out of the **bathroom**, **turn left**, and **go on** to the **stairs** and **wait** near the **coat rack**.

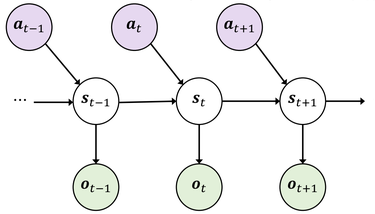


2 Background: Bayesian State Tracking

Given a sequence of observations $\boldsymbol{o}_{1:T}$ and actions $\boldsymbol{a}_{1:T}$, how should we determine the final location $\boldsymbol{s}_T$?

Use **Bayes filter** to estimate probability distribution over latent state $\boldsymbol{s}_T$ given $\boldsymbol{o}_{1:T}$ and $\boldsymbol{a}_{1:T}$.

i.e. at each time step t, compute $bel(\boldsymbol{s}_t) = p(\boldsymbol{s}_t|\boldsymbol{a}_{1:t}, \boldsymbol{o}_{1:t})$ also called *belief*.



Motion Update: (Motion Model)

$$\overline{bel}(\boldsymbol{s}_t) = \int p(\boldsymbol{s}_t|\boldsymbol{s}_{t-1}, \boldsymbol{a}_t)\, bel(\boldsymbol{s}_{t-1})\, d\boldsymbol{s}_{t-1}$$

Observation Update:

$$bel(\boldsymbol{s}_t) = \eta\, p(\boldsymbol{o}_t|\boldsymbol{s}_t)\, \overline{bel}(\boldsymbol{s}_t)$$

(Observation Model)

Recent work [1-3] show Bayes filter can be embedded into deep neural networks.

3 References

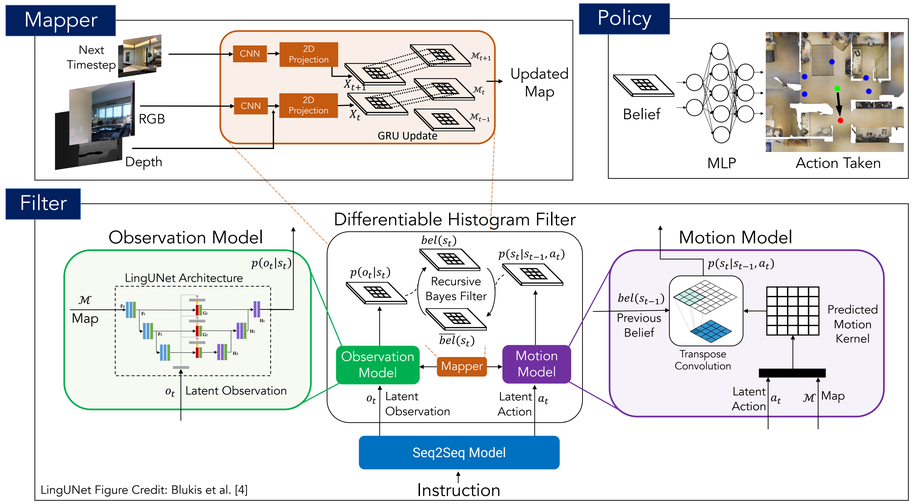
[1] Rico Jonschkowski and Oliver Brock. End-to-end learnable histogram filters. In *In Workshop on Deep Learning for Action and Interaction at the Conference on Neural Information Processing Systems (NIPS)*, 2016.

[2] Rico Jonschkowski, Divyam Rastogi, and Oliver Brock. Differentiable Particle Filters: End-to-End Learning with Algorithmic Priors. In *Proceedings of Robotics: Science and Systems (RSS)*, 2018.

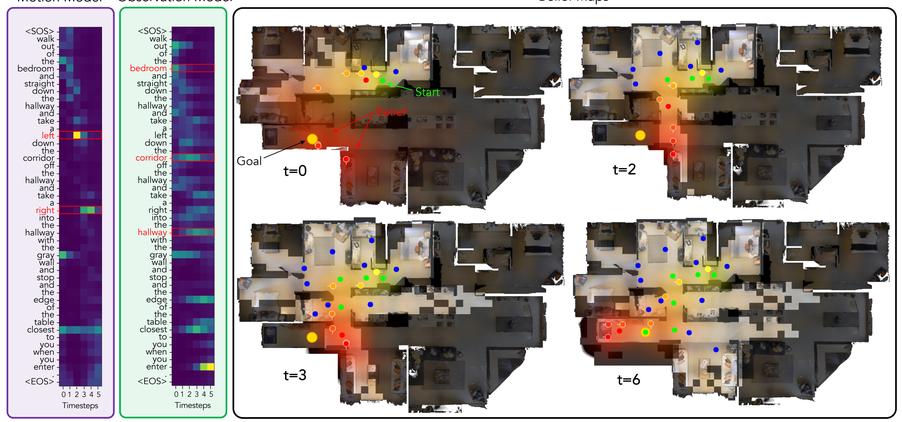
[3] Peter Karkus, David Hsu, and Wee Sun Lee. Particle filter networks with application to visual localization. In *CoRL*, 2018.

[4] Valts Blukis, Dipendra Misra, Ross A Knepper, and Yoav Artzi. Mapping navigation instructions to continuous control actions with position-visitation prediction. In *CoRL*, 2018.

4 Agent Model

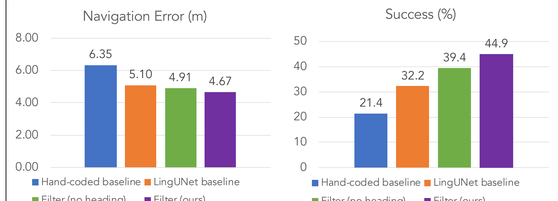


5 Interpretability of Model



6 Results

Goal Location Prediction Task [Mapper + Filter]



Our model outperforms strong baseline (LingUNet [4]).

Vision and Language Navigation (VLN) Task [Mapper + Filter + Policy]

Model	RL	Aug	Val-Seen TL	Val-Seen NE	Val-Seen OS	Val-Seen SR	Val-Seen SPL	Val-Unseen TL	Val-Unseen NE	Val-Unseen OS	Val-Unseen SR	Val-Unseen SPL
Speaker-Follower		✓	-	3.36	0.74	0.66	-	-	6.62	0.45	0.36	-
RCM	✓		10.65	3.53	0.75	0.67	-	11.46	6.09	0.50	0.43	-
Regretful Agent		✓	-	3.23	0.77	0.69	0.63	-	5.32	0.59	0.50	0.41
FAST	✓		-	-	-	-	-	21.1	4.97	-	0.56	0.43
Back Translation	✓	✓	11.0	3.99	-	0.62	0.59	10.7	5.22	-	0.52	0.48
Speaker-Follower			-	4.86	0.63	0.52	-	-	7.07	0.41	0.31	-
Back Translation			10.3	5.39	-	0.48	0.46	9.15	6.25	-	0.44	0.40
Ours			10.15	7.59	0.42	0.34	0.30	9.64	7.20	0.44	0.35	0.31

Our model achieves credible results on the full VLN task.

Training Details

- Trained without Policy.
- Moves towards goal with 50% probability.

#####
FREE SPACE, WHAT TO ADD?
#####

- Samples from Policy with 50% probability, otherwise select GT action.

#####
FREE SPACE, WHAT TO ADD?
#####

7 Conclusion

- Instruction following can be formulated as **Bayesian State Tracking** where model maintains
  - a **semantic map** of the environment,
  - an **explicit probability distribution** over **alternative possible trajectories** in the map.
- Our approach outperforms strong baseline for **goal location prediction**.
- Credible results on the full VLN task without using RL or data augmentation.