An approach to VLN-CE using hierarchy and grounding

VLN-CE setup

- Step size - 0.25m
- Turn Angle - 15°
- Goal Radius - 3m
- Inputs to agent -
  - RGB, Depth and Instruction
- Outputs -
  - One of the 4 actions (STOP, FORWARD, LEFT, RIGHT)

Leave the bedroom and enter the kitchen. Walk forward, And take a left at the couch. Stop in front of the window.

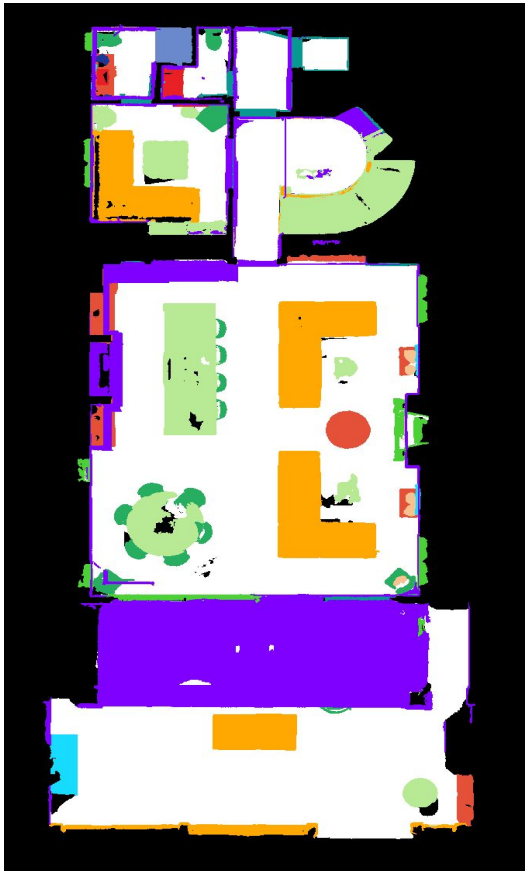
Our Idea

Combining two fairly natural ideas to model a VLN-CE agent

1. Hierarchical Models
    a. VLN-CE paths are long (~60 actions). It's difficult to plan over long horizons. Hence hierarchy might be a better way of modeling it

2. Using a top-down semantic map
    a. Use a top-down semantic map to ground your predictions

Semantic map of scene
(40 categories)



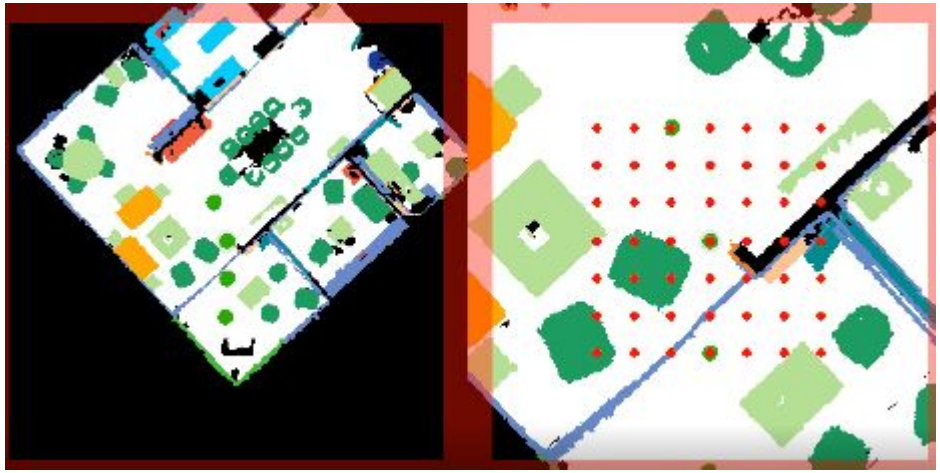
Hierarchical Model

Planner - predicts an intermediate waypoint. It is trained to predict a waypoint *w* forward steps away in the shortest path to the goal

Controller - takes a waypoint as input and navigates towards it for a max of *w* forward steps. Currently using a ‘teleporting’ model

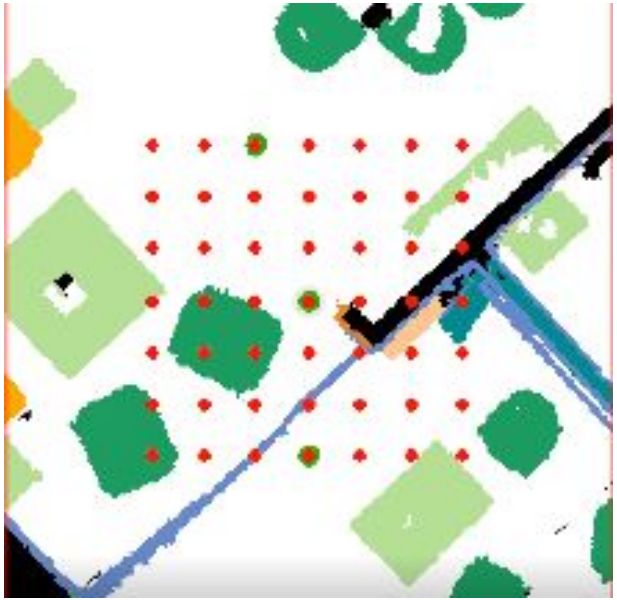
Planner - Input

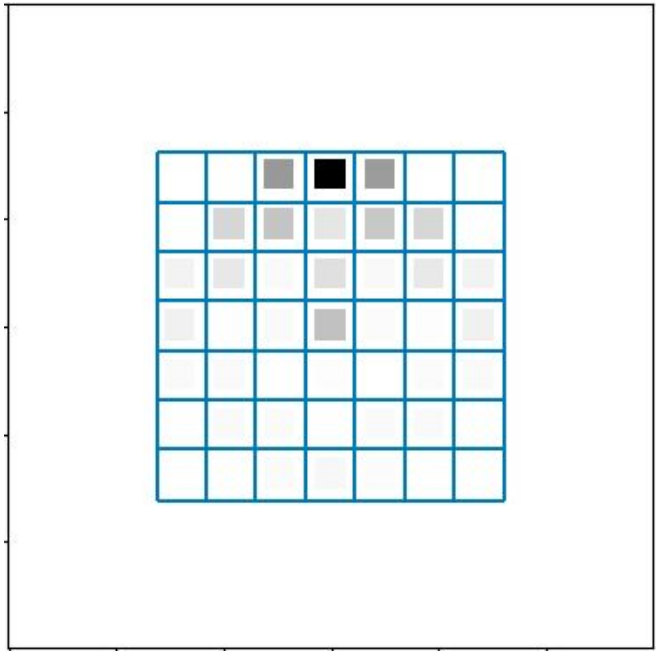
1. RGB
2. Depth
3. Instruction
4. Egocentric Semantic Map
    - Two crops of different resolutions to capture immediate and overall scene
    - Agent at the center and facing upwards
    - Fine crop - Corresponds to a region of 12 forward steps from center till the egde.
    - Coarse crop - Corresponds to a region of 36 forward steps from center till the egde.



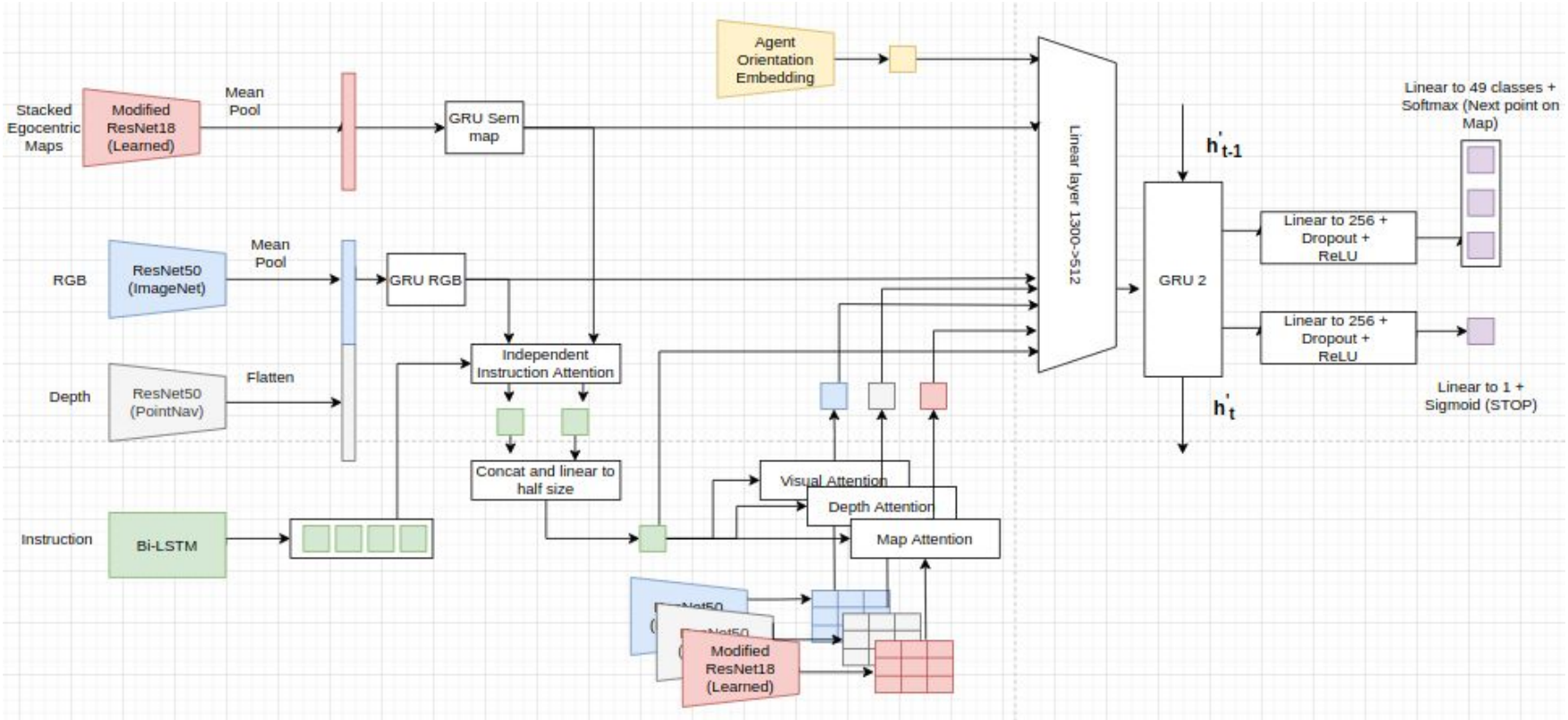
Planner - Output

1. Next waypoint on fine crop (classification across 7*7 grid)
2. Probability of STOP

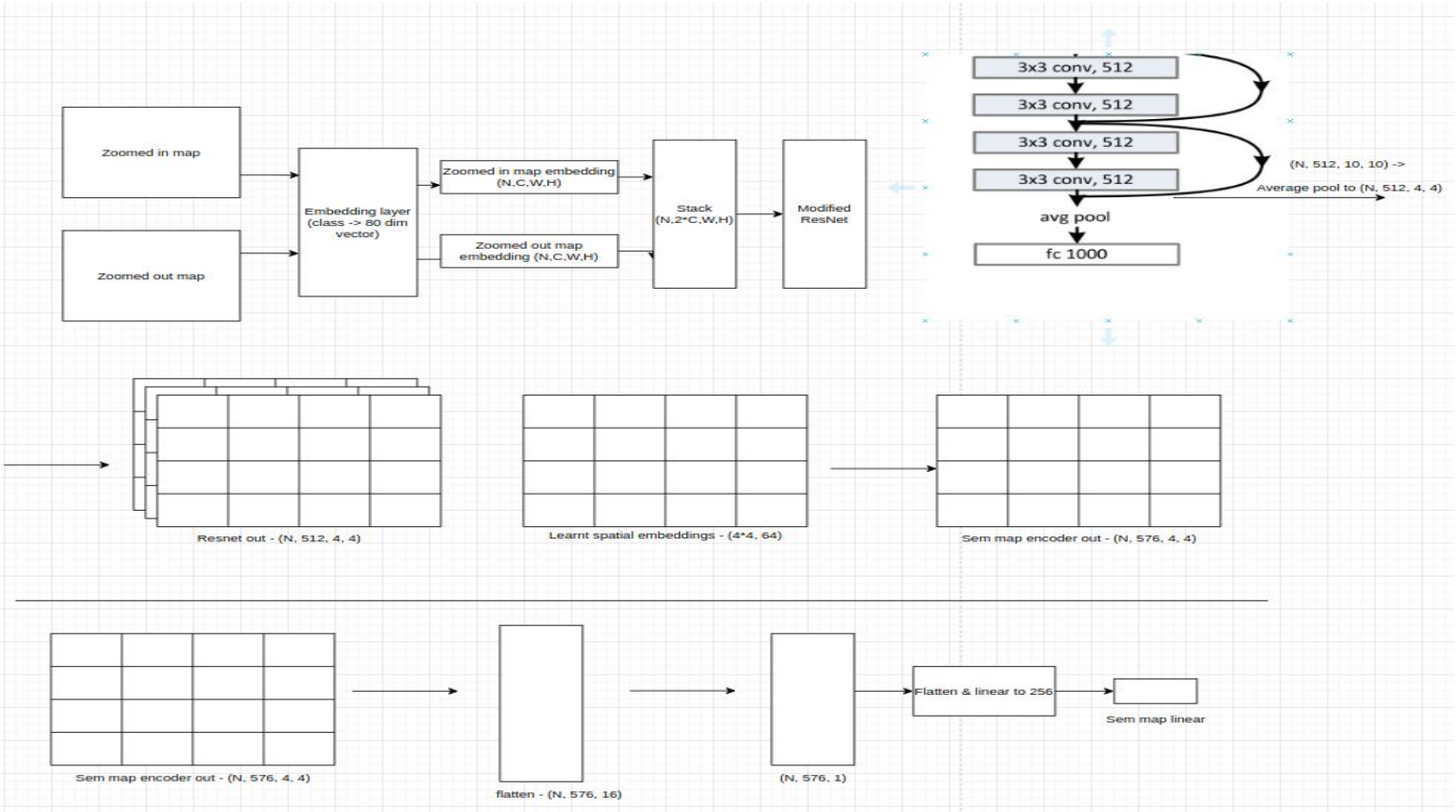




Planner model



Semantic map encoder



Demo

Experiments

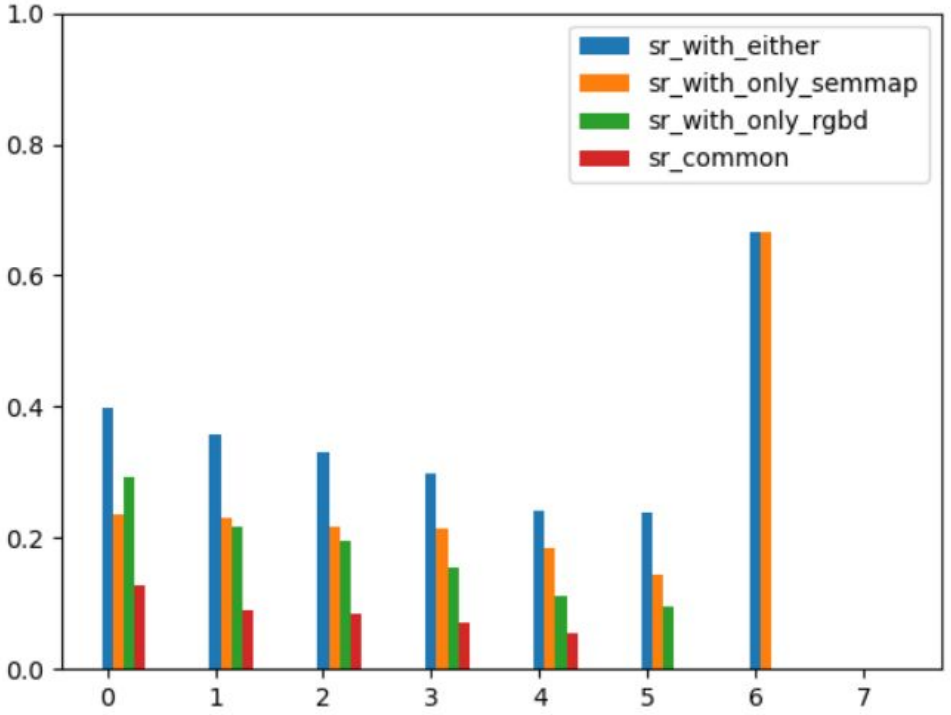
Model	RGB-D	Semantic Map	Floor Map	SR	Remarks
Baseline	Yes	No	No	23	
Our model	Yes	Yes	-	24.60	~7% improvement
Our model without RGBD	No	Yes	-	21.80	
Our model without Sem map	Yes	No	-	20.85	Sem map is more important than RGB-D
Our model with floor map instead of semantic map	Yes	No	Yes	23.44	Just using floor annotations is pretty good too!

Analysis of RGBD model vs Semantic Map model per number of semantic objects in instruction

As no. of objects in the instruction increase, we find a general drop in Success Rate of all models.

But the drop is **less** in the model using only Semantic Map compared to the one using only RGBD. This is an encouraging trend as it means that semantic map is indeed being helpful

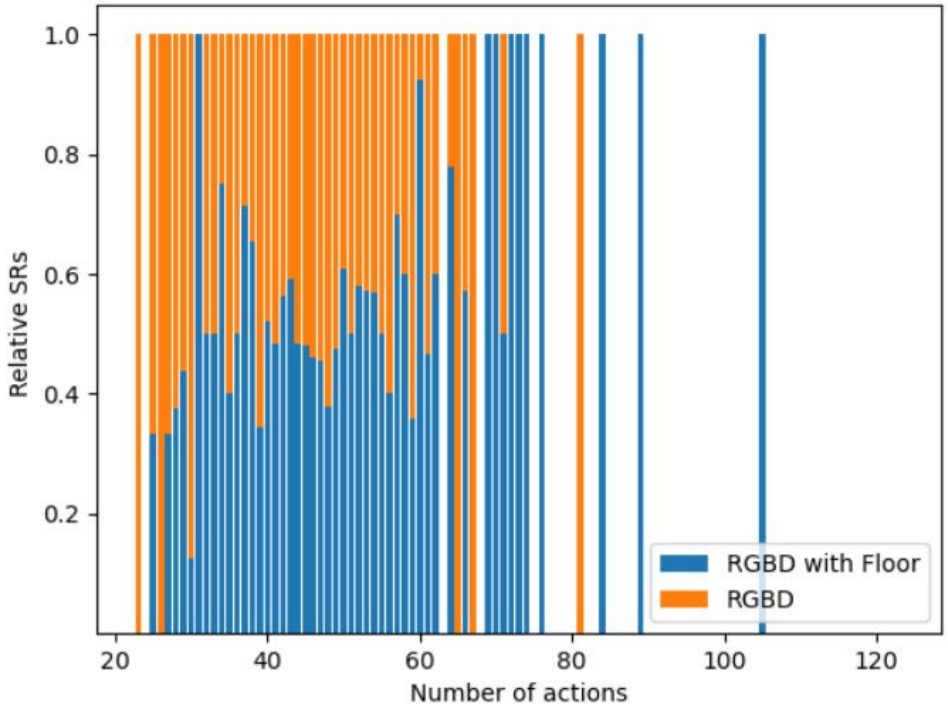
Sem map (i.e,. no RGB-D) - 22.40
RGB-D (i.e., no Sem map) - 21.26
Union of both - 34.63
Intersection of both ~9



Analysis of RGBD model vs RGBD + floor map per number of actions needed to finish episode

As length of episode increases, floor map seems to help more (although noisy indicator due to number of instructions at each value)

SR with RGBD + Floor - 23.81
SR with RGBD - 21.26
Union of both - 35.29
Intersection ~9 (Similar to RGBD case)



Observation 1 - Fusion probably isn’t working well

Model A	Model B	SR A	SR B	SR A & B
RGB-D	RGB-D + Floor	20.85	23.44	9.78
RGB-D	RGB-D + SemMap	20.85	24.6	10.05
RGB-D + Floor	RGB-D + SemMap	23.44	24.6	12.01

Observation 2

This is pretty interesting! We've found out that models trained on the task **without instruction input** work very well. There are potentially a lot of inductive biases on the path that these models can exploit!

Model	SR with Instruction	SR without Instruction
Full Model	24.6	21.21
Model with floor map instead of semantic map	23.05	22.10
Model with no inputs	-	9

Conclusion

- We’ve incorporated hierarchy and grounding to approach the task of VLN-CE
- Using out approach, we showed a 7% improvement over the baseline
- However, different ablations and analysis point some interesting trends
    1. Using just a floor map also results in a positive 2% improvement
    2. Fusion might of different modalities might not be working as well as expected
    3. Models without language instruction work surprisingly well, which indicates that there are a lot of inductive biases encoded in the paths and models