

ALFRED

A Benchmark for Interpreting Grounded Instructions for Everyday Tasks

AskForALFRED.com

[Mohit Shridhar](#)

[Jesse Thomason](#)

[Daniel Gordon](#)

[Yonatan Bisk](#)

[Winson Han](#)

[Roozbeh Mottaghi](#)

[Luke Zettlemoyer](#)

[Dieter Fox](#)

Presented by Goonmeet Bajaj

Agenda

- Overview
- Dataset Curation
- Model Overviews
- Baseline Experiments

Overview

<https://askforalfred.com>

Demo Video



ALFRED

- **ALFRED (Action Learning From Realistic Environments and Directives)**
- New benchmark for:
 - f(natural language instructions and egocentric) → sequences of actions for household tasks
- Aim to shrink the gap between research benchmarks and real-world applications
 - non-reversible state changes
 - connect language to actions, behaviors, and objects in interactive visual environments

Contributions

- New more challenging benchmark
 - High level goal
 - Low level language instructions
- Pixelwise interaction mask of the target object
- Show need for better models
 - < 5% success rate of recent models

Goal: "Rinse off a mug and place it in the coffee maker"



Statistics

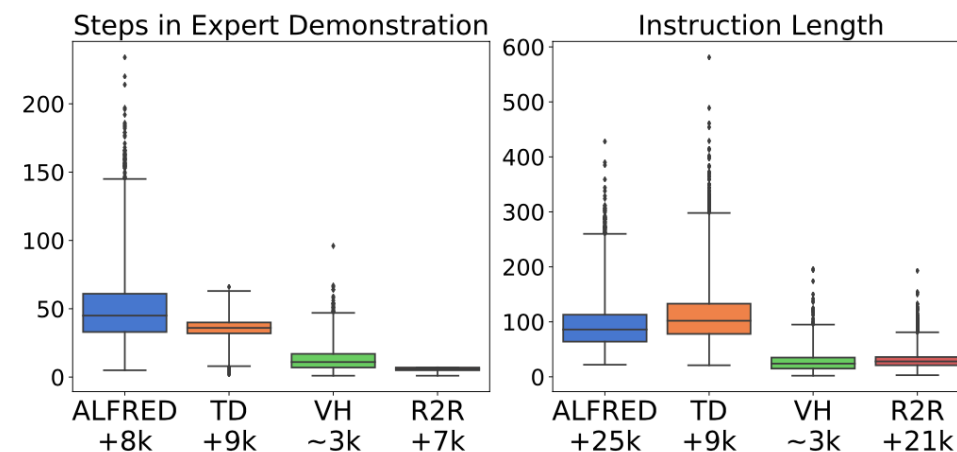
- 120 indoor scenes
- 25,743 English language directives
- 8,055 expert demonstrations averaging 50 steps each
 - 3 expert demonstrations per parameter set (scene, object, task)
- 428,322 image-action pairs
- Generated using the AI2-THOR simulator

	Train	Validation		Test	
		<i>Seen</i>	<i>Unseen</i>	<i>Seen</i>	<i>Unseen</i>
# Annotations	21,023	820	821	1,533	1,529
# Scenes	108	88	4	107	8







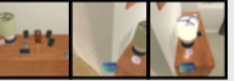
Comparison to prior works

	— Language —		— Virtual Environment —			— Inference —		
	# Human Annotations	Granularity	Visual Quality	Movable Objects	State Changes	Vis. Obs.	Navigation	Interaction
TACoS [43]	17k+	High&Low	Photos	✗	✗	—	—	—
R2R [3]; Touchdown [14]	21k+; 9.3k+	Low	Photos	✗	✗	Ego	Graph	✗
EQA [15]	✗	High	Low	✗	✗	Ego	Discrete	✗
Matterport EQA [55]	✗	High	Photos	✗	✗	Ego	Discrete	✗
IQA [20]	✗	High	High	✗	✓	Ego	Discrete	Discrete
VirtualHome [42]	2.7k+	High&Low	High	✓	✓	3 rd Person	✗	Discrete
VSP [58]	✗	High	High	✓	✓	Ego	✗	Discrete
ALFRED 🧑	25k+	High&Low	High	✓	✓	Ego	Discrete	Discrete + Mask

- More complex scenes with partial environment views
- Navigation, object interactions, state changes



Sample dataset annotations

	Pick & Place	Stack & Place	Pick Two & Place	Clean & Place	Heat & Place	Cool & Place	Examine in Light
item(s)	Book	Fork (in) Cup	Spray Bottle	Dish Sponge	Potato Slice	Egg	Credit Card
receptacle	Desk	Counter Top	Toilet Tank	Cart	Counter Top	Side Table	Desk Lamp
scene #	Bedroom 14	Kitchen 10	Bathroom 2	Bathroom 1	Kitchen 8	Kitchen 21	Bedroom 24
expert demonstration							

	Annotation # 1	Annotation # 2	Annotation # 3
Goals	Put a clean sponge on a metal rack.	Place a clean sponge on the drying rack	Put a rinsed out sponge on the drying rack
Instructions	Go to the left and face the faucet side of the bath tub. Pick up left most green sponge from the bath tub. Turn around and go to the sink. Put the sponge in the sink. Turn on then turn off the water. Take the sponge from the sink. Go to the metal bar rack to the left. Put the sponge on the top rack to the left of the lotion bottle.	Turn around and walk over to the bathtub on the left. Grab the sponge out of the bathtub. Turn around and walk to the sink ahead. Rinse the sponge out in the sink. Move to the left a bit and face the drying rack in the corner of the room. Place the sponge on the drying rack.	Walk forwards a bit and turn left to face the bathtub. Grab a sponge out of the bathtub. Turn around and walk forwards to the sink. Rinse the sponge out in the sink and pick it up again. Turn left to walk a bit, then face the drying rack. Put the sponge on the drying rack.

Dataset Curations

2 Parts - Expert demonstrations & Language directives

Expert demonstrations

- Agent ego-centric view
- Time steps that span actions
- Target object interaction masks (for manipulation actions)
- Designed using the FF planner & Planning Domain Definition Language (PDDL) rules
 - Allows for object positions & object states
- Task parameters: scene, object, task
- Navigation actions:
 - Agent movement
 - Camera rotation
- Manipulation actions:
 - picking and placing objects
 - opening and closing objects
 - turning appliances on and off

2 Parts: Language directives

Language directives

- Consists of high-level goal and low-level instructions
- 3 AMTs per expert demonstration
- Expert demonstrations video and video segments for each action
- AMT task designed using the PDDL plan
 - watches video
 - writes low-level *instructions* for each highlighted sub-goal segment
 - write a high-level *goal* that summarizes the task
- 2 other AMTs validate the directives

Planning Domain Definition Language (PDDL)

- Standard encoding language for planning tasks
- Components of a PDDL planning task:
 - **Objects:** Things in the world that interest us
 - **Predicates:** Properties of objects that we are interested in; can be *true* or *false*
 - **Initial state:** State of the world at time step 0
 - **Goal specification:** Things that we want to be true
 - **Actions/Operators:** Ways of changing the state of the world

<https://www.cs.toronto.edu/~sheila/2542/s14/A1/introtopddl2.pdf>

```

['plan'] = {'high_pddl':
    ...,
    ["high_idx": 4,                (high-level subgoal index)
     "discrete_action":
       {"action": "PutObject",      (discrete high-level action)
        "args": ["bread", "microwave"], (discrete params)
        "planner_action": <PDDL_ACTION> ], (PDDL action)
     ...],

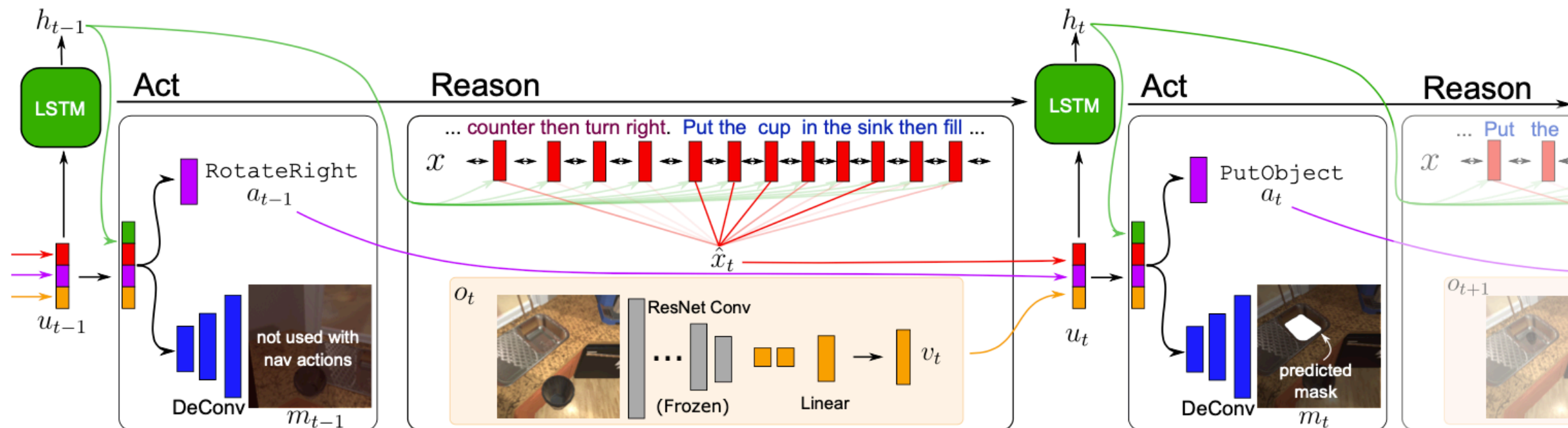
    'low_actions':
    ...,
    ["high_idx": 1,                (high-level subgoal index)
     "discrete_action":
       {"action": "PickupObject",   (discrete low-level action)
        "args":
          {"bbox": [180, 346, 332, 421]} (bounding box for interact action)
          "mask": [0, 0, ... 1, 1]},     (compressed pixel mask for interact action)
        "api_action": <API_CMD> ],      (THOR API command for replay)
     ...],
  }

```

Model Overviews

Seq2Seq, Seq2Seq + Progress Monitoring

Model Overview



Seq2Seq Interactive Agent

- Consists of CNN, Bi-LSTM Encoder, & LSTM Decoder
- Visual features encoded by ResNet-18 CNN
 - Output of last convolution layer embedded using two more convolution layers & a fully-connected layer
 - T observations: encoded as $V = \langle v_1, v_2, \dots, v_T \rangle$
 - v_t = visual feature vector at time t
- Input language (directives) encoded with Bi-LSTM
 - natural language goal: $\bar{G} = \langle g_1, g_2, \dots, g_{L_g} \rangle$
 - step-by- step instructions: $\bar{S} = \langle s_1, s_2 \dots s_{L_s} \rangle$
 - Input seq: $\langle g_1, g_2, \dots, g_{L_g}, \text{<SEP>}, s_1, s_2 \dots s_{L_s} \rangle$
 - Output encoding for each word in input : $\{x_1, x_2, \dots, x_{L_g+L_s}\}$

Seq2Seq Interactive Agent

$$z_t = (W_x h_{t-1})^\top x,$$

$$\alpha_t = \text{Softmax}(z_t),$$

$$\hat{x}_t = \alpha_t^\top x$$

- Soft attention on language features
 - Conditioned on decoder hidden state from last step
 - W = learnable parameters of fully connected layer
 - z vector of scalar values, apply SoftMax to get attention distribution
 - \hat{x} = weighted sum of x over the learned attention distribution
- Action LSTM decoder infers a low-level instruction
 - Input at each time step: visual & language features, and prior action vector
 - Output: new hidden state
 - New hidden state used to learn next attended language feature

$$u_t = [v_t; \hat{x}_t; a_{t-1}],$$

$$h_t = \text{LSTM}(u_t, h_{t-1})$$

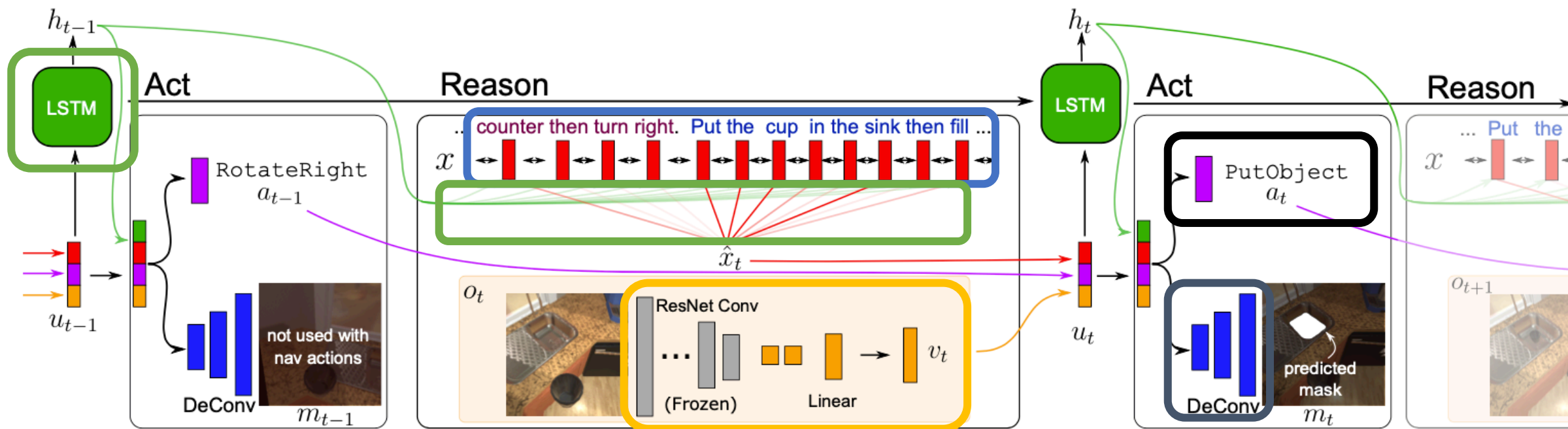
Seq2Seq Interactive Agent

- Target object & action mask prediction
 - Limit to possible 13 action
 - 5 navigation actions: MoveAhead, RotateRight/Left, LookUp/Down
 - 7 interaction actions: Pickup, Put, Open, Close, ToggleOn/Off, Slice
 - 1 stop action
 - Pixelwise mask for interaction actions for identifying the target object
 - Hidden state, input features used in a deconvolution network
 - Softmax cross entropy loss

$$a_t = \operatorname{argmax} (W_a [h_t; u_t]) ,$$

$$m_t = \sigma (\mathbf{deconv} [h_t; u_t])$$

Model Overview



Seq2Seq + Progress Monitors (PM)

- Ma et al. navigation tasks benefit from keeping an internal estimate of their progress
- ALFRED as long sequences of instructions
- Progress monitoring learns the utility of each state while completing the task
- Help distinguish between visually similar states
- Supervision comes from t/T , ration of current time step and total length of expert demonstration

$$p_t = \sigma (W_p [h_t; u_t])$$

Seq2Seq + Progress Monitors (PM)

- Agent is also trained to predict the # of completed subgoal
- Sub-goals are calculated using expert demonstration segments
- Learn alignment between video and language directives
- Supervision comes from c_t/C , ration of current time step and total length of expert demonstration

$$c_t = \sigma (W_c [h_t; u_t]).$$

Baseline Experiments

Evaluation Conditions

- Task Success Metric
 - 1 if the object positions and state changes correspond correctly to the task goal-conditions at the end of the action sequence
 - 0 otherwise
 - “Put a hot potato slice on the counter”
 - succeeds if any *potato slice* object has changed to the *heated* state and is resting on any *countertop* surface
- Goal Success Metric
 - ratio of goal-conditions completed at the end of an episode to those necessary to have finished a task
 - agent slices a *potato*, then moves a slice to the countertop without heating it, then the goal-condition success score is $2/4 = 50\%$.

Evaluation Conditions

$$p_s = s \times \frac{L^*}{\max(L^*, \hat{L})}$$

- Task & Goal Success Path Weighted Metrics
 - Considers the length of the expert demonstration
 - PDDL solver does not guarantee optimality but they are efficient (no exploration)
 - L_star = # of steps in the expert demonstration
 - L_hat = # of steps the model takes during training task
- Sub-goal evaluation
 - Test the ability of a model to accomplish the next sub-goal conditioned on the preceding expert sequence
 - “Put a hot potato slice on the counter”
 - evaluate the sub-goal of navigating to the potato after using the expert demonstration to navigate to and pick up a *knife*

Results

Model	Validation			
	<i>Seen</i>		<i>Unseen</i>	
	Task	Goal-Cond	Task	Goal-Cond
NO LANGUAGE	0.0 (0.0)	5.9 (3.4)	0.0 (0.0)	6.5 (4.7)
NO VISION	0.0 (0.0)	5.7 (4.7)	0.0 (0.0)	6.8 (6.0)
GOAL-ONLY	0.1 (0.0)	6.5 (4.3)	0.0 (0.0)	6.8 (5.0)
INSTRUCTIONS-ONLY	2.3 (1.1)	9.4 (6.1)	0.0 (0.0)	7.0 (4.9)
SEQ2SEQ	2.4 (1.1)	9.4 (5.7)	0.1 (0.0)	6.8 (4.7)
+ PM PROGRESS-ONLY	2.1 (1.1)	8.7 (5.6)	0.0 (0.0)	6.9 (5.0)
+ PM SUBGOAL-ONLY	2.1 (1.2)	9.6 (5.5)	0.0 (0.0)	6.6 (4.6)
+ PM Both	3.7 (2.1)	10.0 (7.0)	0.0 (0.0)	6.9 (5.1)
HUMAN	-	-	-	-

	Test			
	<i>Seen</i>		<i>Unseen</i>	
	Task	Goal-Cond	Task	Goal-Cond
	0.2 (0.0)	5.0 (3.2)	0.2 (0.0)	6.6 (4.0)
	0.0 (0.0)	3.9 (3.2)	0.2 (0.1)	6.6 (4.6)
	0.1 (0.1)	5.0 (3.7)	0.2 (0.0)	6.9 (4.4)
	2.7 (1.4)	8.2 (5.5)	0.5 (0.2)	7.2 (4.6)
	2.1 (1.0)	7.4 (4.7)	0.5 (0.2)	7.1 (4.5)
	3.0 (1.7)	8.0 (5.5)	0.3 (0.1)	7.3 (4.5)
	3.8 (1.7)	8.9 (5.6)	0.5 (0.2)	7.1 (4.5)
	4.0 (2.0)	9.4 (6.3)	0.4 (0.1)	7.0 (4.3)
	-	-	91.0 (85.8)	94.5 (87.6)

Sub-Goal Evaluation

Sub-Goal Ablations - Validation										
Model		<i>Goto</i>	<i>Pickup</i>	<i>Put</i>	<i>Cool</i>	<i>Heat</i>	<i>Clean</i>	<i>Slice</i>	<i>Toggle</i>	Avg.
<i>Seen</i>	No Lang	28	22	71	89	87	64	19	90	59
	S2S	49	32	80	87	85	82	23	97	67
	S2S + PM	51	32	81	88	85	81	25	100	68
<i>Unseen</i>	No Lang	17	9	31	75	86	13	8	4	30
	S2S	21	20	51	94	88	21	14	54	45
	S2S + PM	22	21	46	92	89	57	12	32	46