A Review of "Learning to Interpret Natural Language Navigation Instructions from Observations"

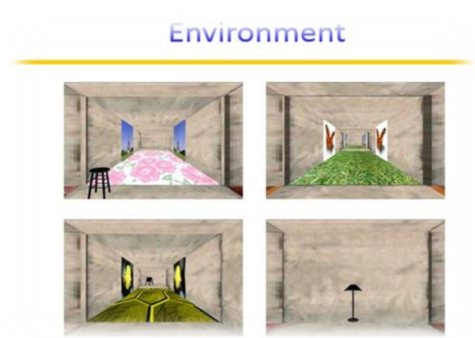
- *A.U.S.S Pradeep (10001),*
February 26, 2013.

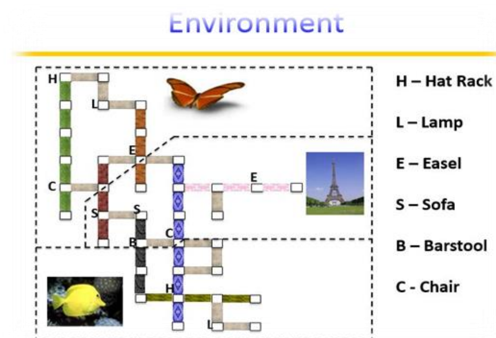
Abstract:

The Main objective is to build a system that can interpret and follow free-form Natural – language instructions to move to a desired location based on the way humans follow sample instructions, with no prior linguistic knowledge: syntactic, semantic, or lexical.

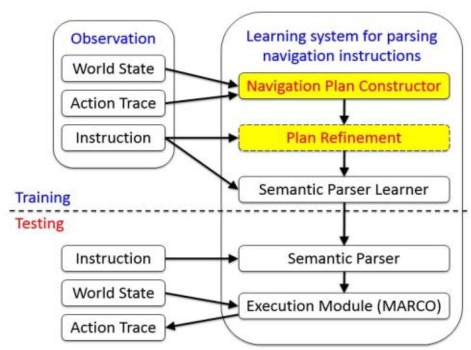
Introduction:

Formally, the system is given training data in the form:{(e_1,a_1,w_1), (e_2,a_2,w_2),...,(e_n , a_n , w_n)}, where e_i is a natural language instruction, a_i is an observed action sequence, and w_i is a description of the current state of the world including the patterns of the floors and walls and positions of any objects. The goal is then to build a system that can produce the correct a_j given a previously unseen (e_j , w_j) pair. But for the purpose of training and testing the data and environments of three different virtual worlds consisting of interconnecting hallways is used.



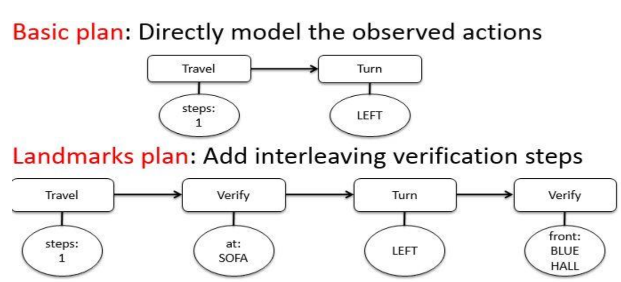


Contributions:



Given a particular position (e_i , a_i , w_i),a particular plan p_i is designed based on the action sequence a_i and the understanding of the state of the world w_i .The pair (e_i , p_i),is then used as learning data for a 'Semantic Parser'. During testing, the parser maps new instructions e_j to formal navigation plans p_j which is carried out by the execution module (MARCO).

- Marco, was an agent that follows free-form, natural language route instructions by representing and executing a sequence of compound action specifications that model which actions to take under which conditions.
- The paper describes about the difference between 'Basic plan' approach and 'Landmarks plan'.
- To remove the extraneous information they employ a 'Plan Refinement', which first learns the meaning of short phrases and words and uses the learned 'Lexicon' to remove parts of plans unrelated to instructions.



- The algorithm collects all plans ‘g’ that occur along with a phrase ‘w’ and takes intersections of all possible pairs of meanings. Further it ranks the entries by scoring functions.

$$Score(w, g) = p(g|w) - p(g|\neg w)$$

  Intuitively, the score measures how much more likely a graph ‘g’ occurs when ‘w’ is present compared to when it is not present.
- For the ‘Learning Semantics Parser’ **KRISP (Kernel-based Robust Interpretation for Se-mantic Parsing)** (Kate and Mooney, 2006) is used which is a supervised learning system for semantic parsing which takes NL sentences paired with their MRs as training data .KRISP trains the classifiers used in semantic parsing iteratively.

Experimental Results:

	Precision	Recall	F1
Basic plans	81.46	55.88	66.27
Landmarks plans	45.42	85.46	59.29
Refined landmarks plans	78.54	78.10	78.32

- By testing how the system infers the correct navigation plans using partial parse accuracy as metric. Compared to basic plans, landmark plans have better recall but considerably lower precision. However the lexicon refined plans retain both high precision and high recall.
- And by testing the end-to-end execution of how well the system can perform the overall navigation task on a strict metric of successful finishes, the following results are obtained.

	Single Sentences	Paragraphs
Simple Generative Model	11.08%	2.15%
Basic Plans	56.99%	13.99%
Landmarks Plans	21.95%	2.66%
Refined Landmarks Plans	54.40%	16.18%
Human Annotated Plans	58.29%	26.15%
MARCO	77.87%	55.69%
Human Followers	N/A	69.64%

- The better performance of the basic plans on the single-sentences task shows that for these shorter instructions, directly modeling the low-level actions is often sufficient.
- Landmarks are useful in recovering from small mistakes in parsing and hence the system using ‘refined landmark plans’ performed better among the first three models for complete instructions test case.
- MARCO was fully manually engineered for this environment and hand-tuned on this data to achieve the best performance.

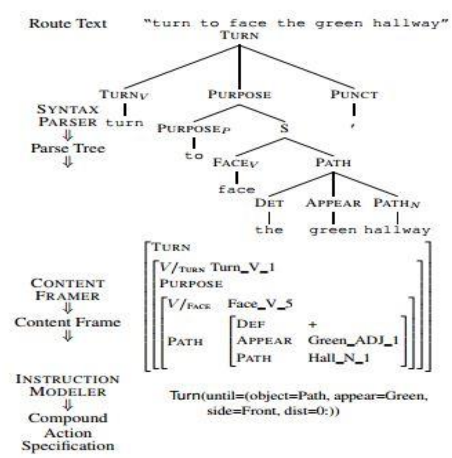


Figure 1: MARCO linguistic modules modeling a route instruction text (Top) through the syntactic verb argument and phrase structure (Mid-Top), the surface semantics frame (Mid-Bottom), and the imperative semantics of which action to take under a minimal model of the context (Bottom).

Conclusion:

This paper showed a novel system that learns a semantic parser for interpreting navigation instructions by simply observing the actions of human followers without using any prior linguistic knowledge or direct supervision.

But one of its shortcomings was that a mistake committed in earlier stage propagated to the later stages and hence a built in feedback loop working iteratively can improve the performance even more.

References:

- [1]"Semi-Supervised Learning for Semantic Parsingusing Support Vector Machines”- Rohit J. Kate and Raymond J. Mooney.
- [2]Matt MacMahon, Brian Stankiewicz & Benjamin Kuipers 'Walk the talk: Connecting language, knowledge, and action in route instructions.'' National Conference on Artificial Intelligence (AAAI-06).
- [3] ‘Learning to Interpret Natural Language Navigation Instructions from Observations’ by David L.Chen, http://www.cs.utexas.edu/users/ml/clamp/navigation/