**OPEX: A Large Language Model-Powered Framework for Embodied Instruction Following**

Haochen Shi  
Université de Montréal & Mila  
Montréal, Canada  
haochen.shi@umontreal.ca

Zhiyuan Sun  
Université de Montréal & Mila  
Montréal, Canada  
zhiyuan.sun@umontreal.ca

Marc-Alexandre Côté  
Microsoft Research  
Montréal, Canada  
macote@microsoft.com

Xingdi Yuan  
Microsoft Research  
Montréal, Canada  
eric.yuan@microsoft.com

Bang Liu  
Université de Montréal & Mila  
Montréal, Canada  
bang.liu@umontreal.ca

**ABSTRACT**

Embodied Instruction Following (EIF) is crucial for understanding natural language in a practical context, requiring agents to follow verbal instructions for complex tasks. Traditionally, EIF relies heavily on expert annotations for learning, which are costly and sometimes unattainable. Recent research shows Large Language Models (LLMs) can use their reasoning ability to help in EIF with minimal examples, but applying LLMs directly faces issues like hallucinations and partially observable environment. To bridge the gap, we introduce OPEX, a new LLM-based method for EIF that needs far less specific data. OPEX uses three LLMs for different roles: observing to gather environment data, planning by breaking down instructions, and executing tasks with learned skills. Our tests reveal OPEX significantly outperforms the FILM baseline, with 90% less training data for planning tasks and achieving up to 38% performance gain when FILM is trained on identical data.

**KEYWORDS**

Embodied Instruction Following; Language Grounding; Large Language Models; Grounded Planning; In Context Learning

ACM Reference Format:


**1 INTRODUCTION**

The creation of autonomous agents requires integrating extensive planning with precise execution, a challenge that deep learning advancements are helping to overcome [1, 7, 8, 11]. Embodied Instruction Following (EIF) has become a key area of focus, necessitating additional methods to follow natural language instructions through egocentric observations [19]. Traditional EIF methods rely heavily on expert annotations, which are costly and sometimes impractical. Large Language Models (LLMs) present a promising solution, trained on extensive data to exhibit common-sense reasoning [5, 16, 21, 22], but direct application to EIF faces challenges like environmental unpredictability and the need for adaptation.

To address these issues, we introduce OPEX (Observer & Planner & Executor), a novel LLM-centric framework for EIF that dynamically integrates planning and action. The Planner uses LLMs for task decomposition, the Observer updates with environmental feedback, and the Executor translates the plans into actionable steps, using a skill set to guide the agent in its tasks. OPEX demonstrates significant improvements on the ALFRED benchmark [19], achieving over 10% absolute performance gains over the baseline FILM [11], requiring 90% less training data. Besides, it achieves up to 38% absolute performance gain when FILM is trained on identical data.

**2 THE OPEX FRAMEWORK**

The OPEX framework introduces a novel approach for Embodied Instruction Following (EIF) with a focus on dynamic task planning and grounding, utilizing Large Language Models (LLMs) for enhanced efficiency and adaptability. Unlike previous methods that depend heavily on static plans and supervised learning, OPEX leverages the reasoning capabilities of LLMs to dynamically decompose tasks, improve grounding, and address the sparse reward problem in EIF without extensive training data or heuristic rules. As shown in Fig. 1, OPEX consists of six main components: (1) semantic mapping module converting egocentric visual observations into semantic maps (2) An LLM-based planner that decomposes language instructions into subtasks. (3) An LLM-based observer that updates the world state in natural language description. (4) An LLM-based executor selecting skills to complete subtasks. (5) A skill library storing predefined skills for manipulation. (6) A deterministic action policy for converting skills into actions.

**Semantic Mapping Module.** This module creates a 2D semantic map from visual inputs, utilizing UNet [18] for depth mapping and MaskRCNN [6] for instance segmentation, following FILM [11]. To address perceptual noise, a supplementary semantic map $M'_{t}$ is proposed aggregating information over time and enhancing reliability.

**LLM-Based Planner.** The LLM-based planner aims to break down a language instruction into subtasks, leveraging LLMs’ reasoning
Observer's observation guidance.

Executor:

RequireReplan, general idea of the environment.

Action:

RotateLeft, MoveAhead, MoveAhead (Produced by action planner)

Skill (Produced by action planner)

Explore

The LLM-based Observer plays a critical role in the OPEx framework by executing sub-tasks using a pre-defined skill library. Unlike the LLM-based planner, the executor is actively involved in the environment, leveraging feedback to understand dynamics and apply the necessary skills to complete tasks.

The main results are shown in Table 1. Remarkably, OPEx leverages in-context learning with less than 10% of the data used for FILM’s Language Processor training yet achieves more than 10% in SR on both splits under all the settings. Table 2 shows OPEx’s superior performance over FILM in utilizing in-domain data. When FILM is trained on the same data, OPEx demonstrates significant improvements across all metrics.

3 EXPERIMENTS AND DISCUSSION

Experiment Setup. Our approach is evaluated on the ALFRED benchmark [19]. We employ four primary evaluation metrics as established in prior works [11, 19]: Success Rate (SR), Goal Condition (GC), path length weighted SR (PLWSR), and path length weighted GC (PLWGC), with SR in the test unseen split serving as the primary performance indicator.

Acknowledgments

This work is supported by the Canada CIFAR AI Chair Program and the Canada NSERC Discovery Grant (RGPIN-2021-03115).
REFERENCES


