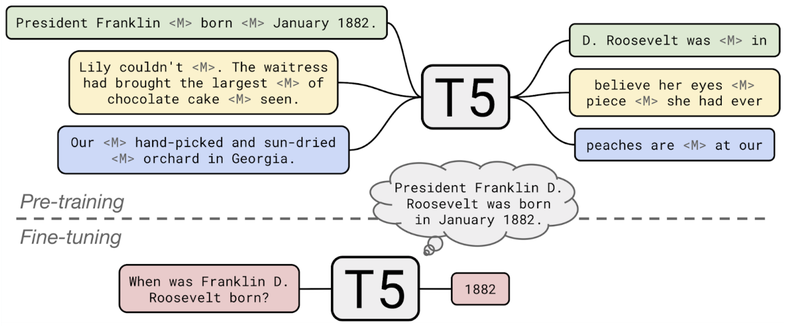
Adaptive Text-to-Command Translation for Robot Navigation: Fine-Tuning T5-Small for Natural Language-Based Control

University of Maryland – College Park – CMSC 723– Natural Language Processing

ABSTRACT

Modern robot navigation systems often rely on manually defined instructions or simple rule-based guidance. This work focuses on fine-tuning a T5-small language model to translate natural language instructions into structured robot navigation commands. By training on a custom dataset of over 24581 complex instruction sequences, we enable a robot to interpret and follow arbitrary directives specifying a sequence of colored battery checkpoints. Our results show a significant increase in accuracy and robustness compared to the baseline pre-trained T5-small model. This approach has potential for broad applications in human-robot interaction, automated delivery systems, and exploration tasks, improving flexibility and autonomy in robotic navigation.



Hugging Face PyTorch

OBJECTIVES

Natural Language Understanding:

Enable robots to comprehend complex, human-like navigation instructions.

Sequence-to-Sequence Learning:

Fine-tune T5-small to produce clear, ordered action sequences **(e.g., "green; red; blue ...").**

Robustness and Scalability:

Achieve high accuracy across simple to complex instructions, ensuring robust performance for diverse real-world scenarios.

ROS Integration:

Seamlessly integrate the model's outputs with the ROS framework for dynamic, simulation-based navigation tasks.

```
--- First prompt ---
Input (with format request): Go to the green battery first, then go to the red battery, followed by the orange battery,
blue battery and purple battery.
Just give me the color order in this way: Go to [color1]; Go to [color2]
Output: the color order: Go to [color1]; Go to [color2]; Go to [color2]; Go to [color2]; Go to [color1]; Go to [color2]

Input (without format request): Go to the green battery first, then go to the red battery, followed by the orange battery,
blue battery and purple battery.
Output: Gehen Sie zu der green battery, dann zu der red battery, followed by orange battery, blue battery and purple battery.
```

Few-shot learning sequence on baseline

```
100%|██████████| 299/299 [07:26<00:00,  1.49s/it]
Sequence Accuracy: 0.0000
Position Accuracy: 0.0000
Total sequences evaluated: 4780
Total positions evaluated: 22620
```

Few-shot Learning Results (Base-line) METRICS

```
Input: Before going to purple battery, start your task at green battery and
make sure to visit blue battery along the way. Then go to the red battery,
and wrap up at the orange battery.
Output: green; blue; red; orange; purple
```

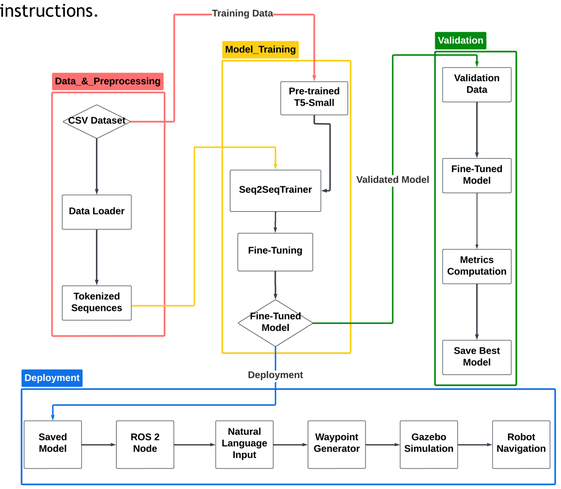
LoRA based Fine-tuned sequence outputs

METHODOLOGIES

•**Data Source:** The Dataset consists of 24581 sentences of instructions for robot navigation, generated with the use of 249 skeleton sentences with placeholders for batteries 1 to 5. The skeleton sentences includes different combinations of sentences consisting of sequences of 1 battery to 5 batteries. This skeleton is then fed into permutations to generate sentences of every possible color order leading to the final dataset as mentioned above. The Dataset is further divided into Training and Validation Dataset which is later fed into the model to prevent **data leak**.

•**Preprocessing:** Tokenization and truncation to a max length of 128 tokens.

•**Fine-Tuning:** Training on the custom dataset with a learning rate of 2e-5, 25 epochs, and weight decay of 0.01. Gradient-based optimization to align model outputs with desired structured instructions.



RESULTS

•**Baseline (Pre-trained T5-small):**

- Struggled even with simple instructions; often repeated input text without generating correct output.

• **Few-shot Learning:**

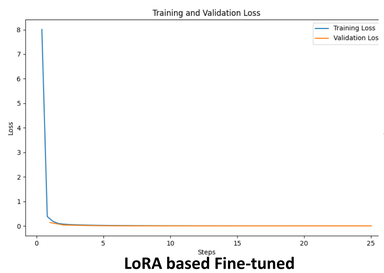
- Provided the same results as the baseline, except for one prompt which is almost identical to the baseline's result

•**Fine-Tuned Model:**

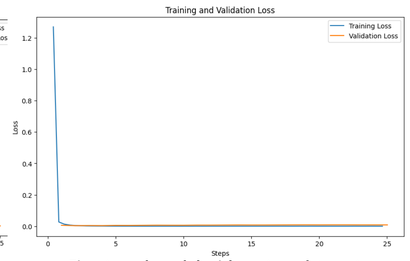
- Achieved ~0.95 sequence accuracy and **~0.97 position accuracy**.
- Correctly identifies and orders battery sequences from complex human-like instructions.
- Improved performance on both simple and complex sentences, demonstrating adaptability.

•**LORA Fine-Tuning:**

- Slight variations in accuracy **(~0.94 sequence accuracy)**, still highly effective.
- Suggests that parameter-efficient fine-tuning methods can maintain strong performance.



LoRA based Fine-tuned



Fine-tuned model with custom data-set

```
  0%|          | 0/299 [00:00<?, ?it/s]
100%|██████████| 299/299 [00:26<00:00, 11.40it/s]
Sequence Accuracy: 0.9498
Position Accuracy: 0.9735
Total sequences evaluated: 4780
Total positions evaluated: 22620
```

Fine-tuned model with custom data-set

```
100%|██████████| 299/299 [02:36<00:00,  1.92it/s]
Sequence Accuracy: 0.9397
Position Accuracy: 0.9735
Total sequences evaluated: 4780
Total positions evaluated: 22620
```

LoRA based Fine-tuned

ROS 2 IMPLEMENTATION

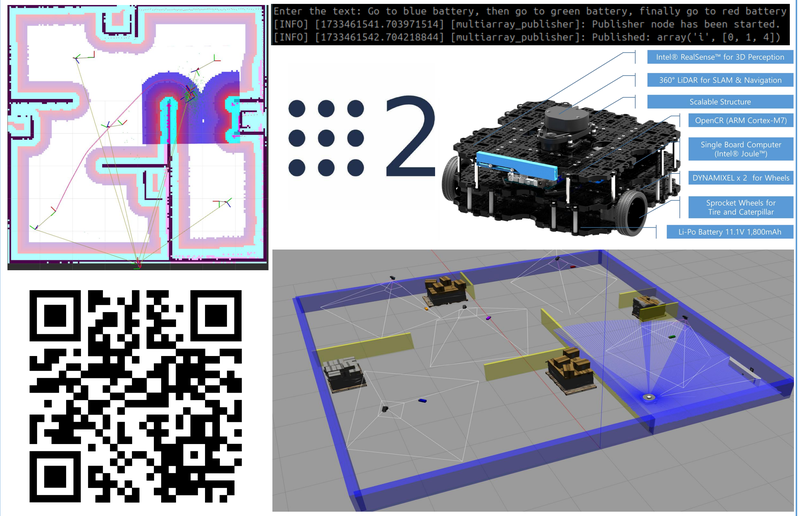
•**Migration to ROS 2 Humble:** Improved tools and libraries, ensuring a modern and reliable development ecosystem.

•**Gazebo Simulation:** Demonstrating the concept by simulating a TurtleBot Waffle in the Gazebo environment, enabling rapid testing and prototyping.

•**SLAM and Localization:** Employing **SLAM techniques** to map and localize the robot, ensuring accurate navigation in dynamic settings.

•**Waypoint Conversion:** Translating the model's output into navigation waypoints, allowing for clear path planning within the simulation.

•**Dynamic Navigation with Nav2: Utilizing the Nav2 library** for flexible waypoint-based navigation, enabling the robot to adapt to changing instructions derived from natural language inputs.



REFERENCES

[1] Raffel, Colin, et al. "Exploring the limits of transfer learning with a unified text-to-text transformer." Journal of
machine learning research 21.140 (2020): 1-67.

[2] https://github.com/google-research/text-to-text-transfer-transformer

[3] Anderson, Peter, et al. "Vision-and-language navigation: Interpreting visually-grounded navigation instructions
in real environments." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[4] https://bringmeaspoon.org

[5] https://github.com/peteanderson80/Matterport3DSimulator

MEMBERS AND CONTACT

•**Adrian Ramsamy** (119066058) | **Email:** aramsamy@umd.edu
•**Akhil Javvadi** (120147517) | **Email:** ajavvadi@umd.edu
•**Sai Jagadeesh Muralikrishnan** (120172243) | **Email:** jagkrish@umd.edu
•**Suhas Nagaraj** (119505373) | **Email:** suhas99@umd.edu
•**Varun Lakshmanan** (120169595) | **Email:** varunl11@umd.edu
•**Vijay Dev Reddy Chevireddi** (119491485) | **Email:** creddy@umd.edu