Literature Review: Vision–and–Language Navigation

Intro

Vision–and–Language Navigation (VLN) is an interdisciplinary field that combines computer vision, natural language processing, and robotics to enable machines to understand and interact with their surroundings through visual and textual inputs. VLN has gained significant attention due to its relevance in a wide range of applications, including autonomous robots, smart home systems, and augmented reality. This literature review aims to provide an overview of the current state of research in VLN, summarize key findings, and identify areas for future investigation.

Real world applications

First of all, it can go a long way in eliminating the need for people to do repetitive daily tasks or dangerous tasks. For example, let's say that at home we can ask a robot to help us get an apple from the kitchen or help us cook. The robot can transmit the real–time picture to us, and then we can use natural language to tell the robot what task should be performed next.

Secondly, VLN is also valuable in theoretical research to explore what embodied AI is, and from this direction to explore the possibility of AGI (Artificial General Intelligence), because it integrates the three modalities of vision, text and action, and can be practically applied to life.

Overview

Vision–and–Language Navigation belongs to embodied AI, which aims to learn by interacting with its surroundings. In other words, an intelligent agent that is able to learn by perceiving a real 3D environment as self–centered as a human. Specifically, the goal of VLN is to allow an agent to explore unseen real–world environments based on natural language commands and visual scenes to achieve specific tasks such as navigating and finding specific objects.

Task: The VLN task consists of three key elements: the agent (a robot that needs to be trained and learn), the oracle (which simulates the role of a human being), and the real environment. the agent can request guidance from the oracle, and the oracle responds. The agent then interacts with the environment and accomplishes specific tasks based on the instructions received and observed. oracle may also help the agent based on the observed environment and agent state.

Example: The robot is randomly placed at a location and then given an instruction related to a distant object such as *'Bring me the bottom picture that is next to the top of stairs on level one'*, the robot needs to interact with the environment according to this instruction and the perceived visual image, the robot needs to find the target object specified by the

instruction. It is worth noting that the target object is not observable at the starting point, which means that the robot must have the common sense and reasoning ability to get to the location where the target is likely to be. Moreover, in the current phase, we only require the robot to find the target object (e.g., by giving the target object's border in the visually perceived image, or by selecting the target object among a set of candidate objects), and we do not need the agent to actually bring the target object back, since the current scene is still non-interactive.

Datasets

Two dimensions:

1. COMMUNICATION COMPLEXITY

We first introduce Communication Complexity. the first level is the **lowest Initial Instruction**: Oracle starts by giving the agent a sentence of Instruction, telling it what task to accomplish, which is the simplest kind. The second is Oracle Guidance, i.e., after receiving the initial task, such as find the apple, if the agent feels confused and it still can't find the apple after searching for a long time, then we tell it that it should turn to the right. the agent needs to have the ability to subsequently continue to understand the natural language. The third one is the most complicated one, which requires it not only to be able to follow up and understand, but also to take the initiative to ask questions in natural language, that is, in the form of Dialogue, which will also be more human-like.

Currently, most of the Dataset are focused on Initial Instructions. there are several reasons for this, one is to create the process of Dataset will be relatively simple, the second Evaluation will also be relatively simple, because if you need to follow up the Oracle Guidance and Dialogue, agent need to Evaluate will have more capabilities.

2. TASK OBJECTIVE

The first one is Fine-grained, where the agent is given very detailed instructions to start with, and it can accomplish the task by following the instructions given to it.

The second is Coarse-grained, where the agent is not given detailed instructions, but only a high-level goal to find the object by itself. For example, if the agent is told to find the mirror in the bedroom, in this case it needs to find the bedroom by itself.

The third one is Navigate and Object Interaction, in order to accomplish the task, the agent not only needs to Navigate, but also needs to do Object Interaction, for example, we ask the agent to go to the kitchen to get some cut apples, but it doesn't find any cut apples, but only whole apples. At this point, Navigation alone is not enough to accomplish this task, it needs to take a knife, to perform the action of interacting with the environment, to cut the apples.

DATASETS EXAMPLES:

Room to Room: Anderson et al. (2018b) created the R2R dataset based on the Matterport 3D simulator (Chang et al., 2017).Specific agents in R2R traverse the edges on the navigation graph by moving through the houses in the simulator, jumping to neighboring nodes that contain panoramic views.

Based on the Matterport3D environment, the authors collected the Room to Room (R2R) dataset, which contains 21,567 open-vocabulary, crowdsourced navigation instructions, with an average length of 29 words. Each instruction describes a trajectory that typically passes through multiple rooms. As shown in Figure 1, the task in question requires agents to follow natural language instructions to navigate to a target location in a previously unseen building

TOUCHDOWN is the first outdoor VLN dataset and the most frequently tested outdoor VLN dataset by researchers today. Almost all of the current outdoor datasets are based on Google Street View because Google has abundant street views that are easy for researchers to extract and use.

REVERIE will initialize the agent to any location in the room, but will not give the agent a specific route, so the agent needs to have a stronger understanding of the room, and even need some Prior Knowledge. e.g., it needs to know where the bedroom is located, if it needs to find a knife, it is usually in the kitchen, and if it needs to find a lamp, it may be in the bedroom. For example, you need to know where the bedrooms are.

Alfred is also a very hot Dataset, it is the first dataset that combines Interaction and Navigation.

CVDN is the first Dialogue Navigation Dataset, where the agent needs to take the initiative to ask questions to solve the Navigation Task. the instructions it receives at the beginning may be very simple, and sometimes it can't rely on the instructions to complete the task, so it needs to keep asking the person how to go during the process, which requires higher linguistic skills for the agent to be able to recognize when to ask questions and what kind of questions to ask. This requires a higher level of language skills, and the agent needs to be able to actively recognize when to ask questions and what kind of questions to ask.

Methods:

1、Representation Learning

The main role of Representation Learning is to help the Agent to get a better understanding of the information.

(1) Pre-training

* CVPR《TowardsLearning a Generic Agent for Vision-and-Language via Pre-training》

Pre-training is currently a very common method in any field, we basically take it as a Common Practice. currently can be used from these dimensions.

The first is unimodal. For example, on top of Language, we directly plug the pre-trained BERT into it, and it will have a very high boost. For multimodal, models like ViLBERT can also

be used on it. The third one is to do pre-training on VLN Domain, there are two ways to do it, the first one is to pre-train from other existing VLN datasets, the second one is to collect or synthesize a large amount of VLN data.

(2) Representation Learning

Semantic Understanding improves the understanding of the input from a semantic point of view. Here it is divided into two directions, the first one is Intra-Modality, which is to enhance from inside the modality and find out what is the pattern inside the modality. For example, to capture important location information from the screen, and then guide the agent where to go next.Inter-Modality is the information between the two modalities, for example, while the screen finds important location information such as staircase, kitchen, table, the related noun vocabulary is found in the language, and matches them together.

(3) Graph Representation

Why Graph is important?VLN involves a lot of explicit information between objects, locations and actions, and if this information is represented in a graph, then it can provide some guidance. Another direction is to build a migration graph. agent discovers a new picture with each step during the exploration of the environment, and using a graph to record all the important information seen can also improve the performance.

(4) Memory Structure

We can also use an additional module to handle the migration history information, VLN involves too much information, for example, all the past Actions, dialog interactions, and the images seen.

(5) Auxiliary Task

Because the VLN Task has a lot of extra information that can help the agent to better understand itself and the environment, this information can be used as an auxiliary task.

2、Action Strategy Learning

Next, we introduce Action Strategy Learning, i.e., after understanding the input information, how to make a correct decision.

(1) Reinforcement Learning
Landmark-RxR: Solving Vision-and-Language Navigation with Fine-Grained Alignment Supervision

The first one is Reinforcement Learning, VLN execution is a dynamic process, given the initial state, the agent needs to perform many actions until the task succeeds or fails.A very important problem in Reinforcement Learning is how to define the Reward.There are different

solutions to this point, such as the most basic is to use the existing There are different solutions for this, such as the most basic one is to use the existing Evaluation Metric, for example, to see what the final success rate is, and then use such a metric to do the reward. the second method is to configure a training model to get the Reward, and for each step, the model will tell us the Reward of the current action from end to end.

(2) Exploration during Navigation

During Navigation, the agent can explore multiple perspectives, and decide which step is better for the agent to move to after looking around the environment.

(3) Navigation Planning

There is another method of Navigation Planning, that is, planning the way to move in advance, the plan can be obtained from both visual and verbal directions. For example, from the visual aspect, you can predict which points in the screen are certain to go, and move in order.

(4) Asking For Help

Taking the initiative to ask for help can help the agent to get out of the dilemma. There are two specific abilities involved here, the first one is that the agent should know when to ask questions, and the second one is how to ask questions in natural language.

3、Data-centric Learning

Data-centric Learning is a very effective method for VLN.

At present, VLN is too complex, involving too much information, first of all, the visual image, then the language information, and action information, all of these information is required to understand, and then output an action, and at the same time, the image can be particularly complex. Under the current situation of extreme data scarcity, Data-centric Learning often plays a very large role.

(1) Data Augmentation

The first and most natural method is Data Augmentation, how to generate more data for training, roughly divided into two ways, the first is Trajectory-Instruction Augmentation, directly generate a path, and then with the language; the other is Environment-Augmentation. The other is Environment Augmentation, which augments the environment first, and then builds more paths in it when there are more virtual environments.

(2) Curriculum Learning

Since Curriculum Learning has been proposed, it has been tried in every field.In VLN, start with simple tasks, train on simple tasks, then train on complex tasks. We can use the number of houses in the migration scenario or the length of natural language instructions to decide the task difficulty.

(3) Multitask Learning

Multitask learning is very common in any field, because different VLN Tasks can help each other, and the implementation is very simple, an agent is used on two datasets at the same time, and then train on these two datasets at the same time.

(4) Instruction Interpretation

Instruction Interpretation, is how to explain Instruction in a different perspective, for example, Instruction is sometimes very long, if it is broken down into many detailed steps, the Agent will be able to better understand and execute.

Existing commonly used Metrics can be categorized into two types, the first one is Goal–Oriented Metrics, in short, is this task successful? The first is Goal–Oriented Metrics, in short, whether the task is successful or not, in which case success is 1 and failure is 0. There are also metrics such as Goal Progress, in which even if it is unsuccessful, the number of steps taken, which is close to the target distance, can be a good metric. another type of metric measures path fidelity, which requires that the paths taken are as similar as possible to the Ground Truth paths.

Evaluation

(1) Trajectory Length (TL): measures the average length of the navigation trajectory.

(2) Navigation Error (NE): is the average value of the shortest path distance between the agent's final position and the target position in meters.

(3) Success Rate (SR): is the percentage of final localizations of the agent that are less than 3 meters away from the target location.

(4) Success weighted by Path Length (SPL): weighs SR against TL. The higher the score, the more efficient the navigation.

Key Points1

Vision–and–Language Navigation (VLN) task is a challenging research field, and its challenge is mainly reflected in the following two aspects:

The multimodal input to the agent is significantly different: this means that the agent needs to simultaneously understand and process information from two different senses, namely **vision (images or videos) and language (textual instructions)**, when performing a task. These two types of information are often different, so the agent needs to combine them to navigate or perform tasks efficiently. This is a complex challenge for computer systems because data from different modalities need to work together.

Training data on new tasks is often limited: **This means that researchers and developers often have limited data to train agents on VLN tasks.** This limits the agent's ability to learn and generalize, as limited data may not be enough to cover all possible situations and scenarios. Therefore, agents need to have strong generalization capabilities and be able to perform new tasks without a large number of examples.

(1) The first to propose pre-training and fine-tuning algorithms for VLN tasks
(2) Trained in a self-supervised learning manner on a large number of image-text-action triples. The pre-training model provides a universal expression of visual environment and language instructions.
(3) Be more effective in new tasks and expand to previously unseen environments.

Partially Observable Markov Decision Process,(POMDP)

Input embedding: the input embedding layer transforms the input (i.e., panoramas and linguistic instructions) into two sequences of features: image-level visual embedding and word-level sentence embedding.

CODING STRUCTURE: The backbone has three main modules, including two unimodal encoders (one for each mode) followed by a transmodal encoder. All modules are based on a multilayer Transformer.

“We introduce two main tasks to pre-train our model: image-attended masked language modeling (MLM) and action prediction (AP).”

“We focus on three downstream VLN tasks based on the Matterport3D simulator. Each task presents different challenges for evaluating the agent. (i) The R2R task is used as an in-domain task; it validates the agent's ability to generalize to unseen environments. (ii) CVDN and HANNA are considered as out-of-domain tasks to study the agent's ability to generalize to new tasks. More specifically, CVDN considers indirect instructions (i.e., dialog history) and HANNA is an interactive RL task.”

Key Points2

The paper titled "Landmark-RxR: Solving Vision-and-Language Navigation with Fine-Grained Alignment Supervision"[1] proposes a novel approach to address the cross-modal alignment challenge in Vision-and-Language Navigation (VLN) task. The authors introduced a human-annotated fine-grained VLN dataset called Landmark-RxR to alleviate weak cross-modal alignment supervision from coarse-grained data. They also investigated the focal-oriented rewards with soft and hard forms to enhance local cross-modal alignment under fine-grained supervision. Moreover, they proposed a re-initialization mechanism that makes metrics insensitive to difficult points, which can cause the agent to deviate from the correct trajectories. The authors' experimental results showed that their agent has superior navigation performance on Landmark-RxR, en-RxR and R2R datasets.

The paper proposes to improve vision and language navigation by augmenting the RxR dataset with corresponded subinstructions and subtrajectories. Using the notion of critical points (points that are more important/helpful in the navigation process), two kinds of focal-oriented rewards (soft and hard) are proposed to encourage the predicted trajectory to be close to the demonstration trajectories at critical points. The authors conduct experiments on R2R and RxR and show that using the collected subinstructions and subtrajectories and proposed losses, the trained agent is able to outperform the baseline RCM model. For the RxR dataset, the training data consisted both of original RxR data as well as an augmented dataset consisting of concatenating the collected subinstruction and subtrajectories together. They also propose a re-initialization mechanism that makes metrics insensitive to difficult points, which can cause the agent to deviate from the correct trajectories. The authors' experimental results show that their agent has superior navigation performance on Landmark-RxR, en-RxR and R2R datasets.

Reference

- Jing Gu, Eliana Stefani, Qi Wu, Jesse Thomason, and Xin Wang. 2022. Vision-and-Language Navigation: A Survey of Tasks, Methods, and Future Directions.
  In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 7606—7623, Dublin, Ireland. Association for Computational Linguistics.
- Hao, W., Li, C., Li, X., Carin, L. and Gao, J. (2020). Towards Learning a Generic Agent for Vision-and-Language Navigation via Pre-training. [online] arXiv.org. doi:https://doi.org/10.48550/arXiv.2002.10638.
- Keji He, Yan Huang, Qi Wu, Jianhua Yang, Dong An, Shuanglin Sima, and Liang Wang. 2021. Landmark- rxr: Solving vision-and-language navigation with fine-grained alignment supervision. In NeurIPS.
- Mahajan, Jay, Hum, Samuel, Ginger, Jeff, and Lane, H. Chad. MineObserver: A Deep Learning Framework for Assessing Natural Language Descriptions of Minecraft Imagery. Retrieved from https://par.nsf.gov/biblio/10343546. The International FLAIRS Conference Proceedings 35. Web. doi:10.32473/flairs.v35i.130729.

Wijmans, E., Savva, M., Essa, I., Lee, S., Morcos, A.S. and Batra, D. (2023). Emergence of Maps in the Memories of Blind Navigation Agents. [online] arXiv.org. doi:https://doi.org/10.48550/arXiv.2301.13261.