Language-Aligned Waypoint (LAW) Supervision for Vision-and-Language Navigation in Continuous Environments

Sonia Raychaudhuri[1] **Saim Wani**[2] **Shivansh Patel**[2]
Unnat Jain[3] **Angel X. Chang**[1]
[1]Simon Fraser University [2]IIT Kanpur [3]UIUC
[1]{sraychau,angelx}@sfu.ca
[2]{saimdwani,shivanshpatel35}@gmail.com [3]uj2@illinois.edu
https://3dlg-hcvc.github.io/LAW-VLNCE

Abstract

In the Vision-and-Language Navigation (VLN) task an embodied agent navigates a 3D environment, following natural language instructions. A challenge in this task is how to handle 'off the path' scenarios where an agent veers from a reference path. Prior work supervises the agent with actions based on the shortest path from the agent's location to the goal, but such goal-oriented supervision is often not in alignment with the instruction. Furthermore, the evaluation metrics employed by prior work do not measure how much of a language instruction the agent is able to follow. In this work, we propose a simple and effective language-aligned supervision scheme, and a new metric that measures the number of sub-instructions the agent has completed during navigation.

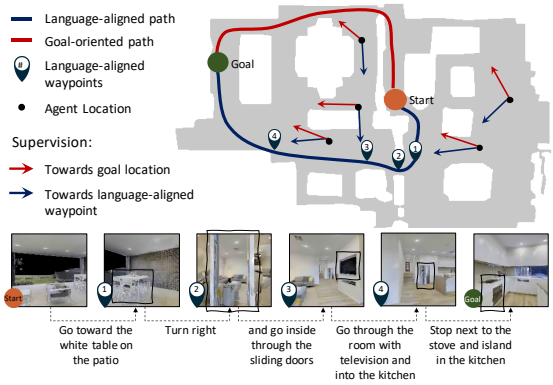

Figure 1: A language-aligned path (blue) in an instruction following task may differ from the shortest path (red) to the goal. Language-aligned supervision (blue arrows) encourages the agent at any given location (dark circles) to move towards the nearest waypoint on the language-aligned path and can hence be a better supervisory signal for instruction following than goal-oriented supervision (red arrows).

1 Introduction

Training agents to navigate in realistic environments based on natural language instructions is a step towards building robots that understand humans and can assist them in their daily chores. Anderson et al. (2018b) introduced the Vision-and-Language Navigation (VLN) task, where an agent navigates a 3D environment to follow natural language instructions. Much of the prior work on VLN assumes a discrete navigation graph (nav-graph), where the agent teleports between graph nodes, both in indoor (Anderson et al., 2018b) and outdoor (Chen et al., 2019; Mehta et al., 2020) settings. Krantz et al. (2020) reformulated the VLN task to a continuous environment (VLN-CE) by lifting the discrete paths to continuous trajectories, bringing the task closer to real-world scenarios.

Krantz et al. (2020) supervised agent training with actions based on the shortest path from the agent's location to the goal, following prior work in VLN (Fried et al., 2018; Tan et al., 2019; Hu et al., 2019; Anderson et al., 2019). However, as Jain et al. (2019) observed, such supervision is goal-oriented and does not always correspond to following the natural language instruction.

Our key idea is that language-aligned supervision is better than goal-oriented supervision, as the path matching the instructions (language-aligned) may differ from the shortest path to the goal (goal-oriented). This is especially true in 'off the path' scenarios (where the agent veers off the reference path prescribed by the instructions). Language-aligned supervision encourages the agent to move towards the nearest waypoint on the language-aligned path at every step and hence supervises the agent to better follow instructions (see Figure 1). In the discrete nav-graph setting, Jain et al. (2019) interleave behavioral cloning and policy gradient training, with a sparse 'fidelity-oriented reward' based on how well each node is covered on the reference path. In contrast, we tackle the VLN-CE setting and propose a simple and effective approach that provides a denser supervisory signal leading

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 4018–4028
November 7–11, 2021.

the agent to the reference path. A dense supervisory signal is especially important for VLN-CE where the episodes have a longer average length of 55.88 steps *vs.* 4-6 nodes in (discrete) VLN. To this end, we conduct experiments investigating the effect of density of waypoint supervision on task performance.

To assess task performance, we complement the commonly employed normalized Dynamic Time Warping (nDTW) metric (Ilharco et al., 2019) with a intuitive *Waypoint Accuracy* metric. Finally, to provide qualitative insights about degree of following instructions, we combine language-aligned waypoints with information about sub-instructions. Our experiments show that our language-aligned supervision trains agents to more closely follow instructions compared to goal-oriented supervision.

2 Related Work

Vision-and-Language Navigation. Since the introduction of the VLN task by Anderson et al. (2018b), there has been a line of work exploring improved models and datasets. The original Room-to-Room (R2R) dataset by Anderson et al. (2018b) provided instructions on a discrete navigation graph (nav-graph), with nodes corresponding to positions of panoramic cameras. Much work focuses on this discrete nav-graph setting, including cross-modal grounding between language instructions and visual observations (Wang et al., 2019), addition of auxiliary progress monitoring (Ma et al., 2019), augmenting training data by re-generating language instructions from trajectories (Fried et al., 2018), and environmental dropout (Tan et al., 2019).

However, these methods fail to achieve similar performance in the more challenging VLN-CE task, where the agent navigates in a continuous 3D simulation environment. Chen et al. (2021) propose a modular approach using topological environment maps for VLN-CE and achieve better results. In concurrent work, Krantz et al. (2021) propose a modular approach to predict waypoints on a panoramic observation space and use a low-level control module to navigate. However, both these works focus on improving the ability of the agent to reach the goal. In this work, we focus on the VLN-CE task and on accurately following the path specified by the instruction.

Instruction Following in VLN. Work in the discrete nav-graph VLN setting has also focused on improving the agent's adherence to given instructions. Anderson et al. (2019) adopt Bayesian state tracking to model what a hypothetical human demonstrator would do when given the instruction, whereas Qi et al. (2020) attends to specific objects and actions mentioned in the instruction. Zhu et al. (2020) train the agent to follow shorter instructions and later generalize to longer instructions through a curriculum-based reinforcement learning approach. Hong et al. (2020) divide language instructions into shorter sub-instructions and enforce a sequential traversal through those sub-instructions. They additionally enrich the Room-to-Room (R2R) dataset (Anderson et al., 2018b) with the sub-instruction-to-sub-path mapping and introduce the Fine-Grained R2R (FG-R2R) dataset.

More closely related to our work is Jain et al. (2019), which introduced a new metric – Coverage weighted by Length Score (CLS), measuring the coverage of the reference path by the agent, and used it as a sparse fidelity-oriented reward for training. However, our work differs from theirs in a number of ways. First, in LAW we *explicitly* supervise the agent to navigate back to the reference path, by dynamically calculating the closest waypoint (on the reference path) for any agent state. In contrast to calculating waypoints, Jain et al. (2019) optimize accumulated rewards, based on the CLS metric. Moreover, we provide dense supervision (at every time step) for the agent to follow the reference path by providing a cross-entropy loss at all steps of the episode, in contrast to the single reward at the end of the episode during stage two of their training. Finally, LAW is an online imitation learning approach, which is simpler to implement and easier to optimize compared to their policy gradient formulation, especially with sparse rewards. Similar to Jain et al. (2019), Ilharco et al. (2019) train one of their agents with a fidelity oriented reward based on nDTW.

3 Approach

Our approach is evaluated on the VLN-CE dataset (Krantz et al., 2020), which is generated by adapting R2R to the Habitat simulator (Savva et al., 2019). It consists of navigational episodes with language instructions and reference paths. The reference paths are constructed by taking the discrete nav-graph nodes corresponding to positions of panoramic cameras (we call these `pano` waypoints, shown as gray circles in Figure 2 top), and taking the shortest geodesic distance between them

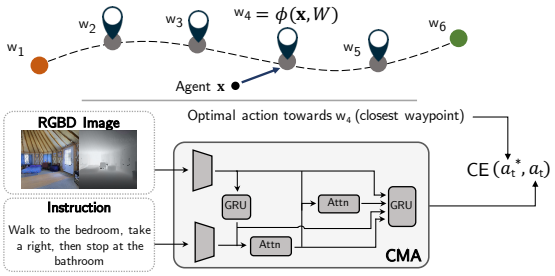

Figure 2: Top: The path from the start (orange) to the goal (green) with grey circle indicating LAW `pano` and the dashed segments indicating LAW `step`. Bottom: We adapt the Cross-Modal Attention (CMA) model (2020) which predicts an action. We optimize the model using language-aligned supervision, which brings it back on the path toward the next waypoint.

to create a ground truth reference path consisting of dense waypoints (`step` waypoints, see dashed path in Figure 2) corresponding to an agent step size of $0.25m$.

We take waypoints from these paths as language-aligned waypoints (LAW) to supervise our agent, in contrast to the goal-oriented supervision in prior work. We interpret our model performance qualitatively, and examine episodes for which the ground-truth language-aligned path (LAW `step`) does not match goal-oriented shortest path (shortest)[1].

Task. The agent is given a natural language instruction, and at each time step t, the agent observes the environment through RGBD image I_t with a $90°$ field-of-view, and takes one of four actions from $\mathcal{A}$: {*Forward*, *Left*, *Right*, *Stop*}. *Left* and *Right* turn the agent by $15°$ and *Forward* moves forward by 0.25m. The *Stop* action indicates that the agent has reached within a threshold distance of the goal.

Model. We adapt the Cross-Modal Attention (CMA) model (see Figure 2) which is shown to perform well on VLN-CE. It consists of two recurrent networks, one encoding a history of the agent state, and another predicting actions based on the attended visual and instruction features (see supplement for details).

Training. We follow the training regime of VLN-CE. It involves two stages: behavior cloning (with teacher-forcing) on the larger augmented dataset to train an initial policy, and then fine-tuning with DAgger (Ross et al., 2011). DAgger trains the model on an aggregated set of all past trajectories, sampling actions from the agent policy. Rather than supervising with the conventional goal-oriented sensor, we supervise with a language-aligned sensor in both the teacher-forcing phase and the DAgger phase. The language-aligned sensor helps bring the agent back on the path to the next waypoint if it wanders off the path (see Figure 2 top).

The training dataset $\mathcal{D} = \{S^{(i)}, W^{(i)}\}$ consists of instructions $S^{(i)}$ and reference path $W^{(i)}$. For each episode $(S, W) \sim \mathcal{D}$ with the agent starting state as x_0, we use cross entropy loss to maximize the log likelihood of the ground-truth action a^* at each time step t: $\mathcal{L}_{\text{CE}}(x_0; \theta) = -\sum_{t=1}^{T} \mathbf{e}_{a^*} \cdot \log \pi_\theta(a_t | I_t, S, x_t, x_0)$. Here, x_t is the 3D position of the agent at time t, $\mathbf{e}_{a^*}$ is the one-hot vector for the ground-truth action a^*, which is defined as $a^* = g(x_t, \phi(x_t, W))$. The set of language-aligned waypoints is $W = \{w_1, ..., w_m\}$. The waypoint in W that is nearest to a 3D position x_t is obtained by $\phi(x_t, W)$. The best action based on the shortest path from a 3D position x_t to w is denoted by $g(x_t, w)$.

4 Experiments

Dataset. We base our work on the VLN-CE dataset (Krantz et al., 2020). The dataset contains 4475 trajectories from Matterport3D (Chang et al., 2017). Each trajectory is described by multiple natural language instructions. The dataset also contains $\sim 150k$ augmented trajectories generated by Tan et al. (2019) adapted to VLN-CE. To qualitatively analyze our model behavior, we use the Fine-Grained R2R (FG-R2R) dataset from Hong et al. 2020. It segments instructions into sub-instructions[2] and maps each sub-instruction to a corresponding sub-path.

Evaluation Metrics. We adopt standard metrics used by prior work (Anderson et al., 2018b,a; Krantz et al., 2020). In the main paper, we report Success Rate (SR), Success weighted by inverse Path Length (SPL), Normalized dynamic-time warping (nDTW), and Success weighted by nDTW (SDTW). Trajectory Length (TL), Navigation Error (NE) and Oracle Success Rate (OS) are reported in the supplement. Since none of the existing metrics directly measure how effectively waypoints are visited by the agent, we introduce *Waypoint Accuracy* (WA) metric. It measures the fraction of waypoints the agent is able to visit correctly (specifically, within 0.5m of the waypoint). This allows the community to intuitively analyze

[1] We find $\sim$6% of the VLN-CE R2R episodes to have nDTW(shortest, LAW `step`) < 0.8 (see supplement)

[2] There are on average 3.1 sub-instructions per instruction

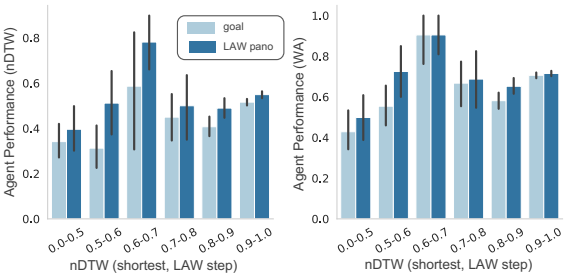

Figure 3: Agent performance binned by nDTW value of reference path to shortest path (95% CI error bars) shows that LAW `pano` performs better than `goal`, especially on lower-range NDTW episodes. This indicates that language-aligned supervision is better suited for the instruction following task.

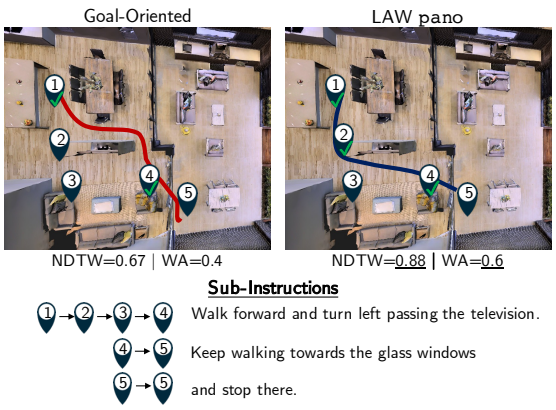

Figure 4: An example episode from R2R unseen split. The agent is able to learn to follow instruction better when supervised with language-aligned path (right) than the goal-oriented path (left). This is reflected in higher nDTW and waypoint accuracy (WA) metrics. Note that WA can be intuitively visualized and interpreted. We also show the mapping of sub-instructions to waypoints utilizing FG-R2R for this episode.

the agent trajectory as we illustrate in Figure 4.

Implementation Details. We implement our agents using PyTorch (Paszke et al., 2019) and the Habitat simulator (Savva et al., 2019). We build our code on top of the VLN-CE codebase[3] and use the same set of hyper-parameters as used in the VLN-CE paper. The first phase of training with teacher forcing on the 150k augmented trajectories took ∼60 hours to train, while the second phase of training with DAgger on the original 4475 trajectories took ∼36 hours over two NVIDIA V100 GPUs.

Ablations. We study the effect of varying the density of language-aligned waypoints on model performance. For all the ablations we use the CMA model described in Section 3. We use LAW # to distinguish among the ablations. On one end of the density spectrum, we have the base model which is supervised with only the goal (LAW#1 or `goal`). On the other end is LAW `step` which refers to the pre-computed dense path from the VLN-CE dataset and can be thought of as the densest supervision available to the agent. In the middle of the spectrum, we have LAW `pano`, which uses the navigational nodes (an average of 6 nodes) from the R2R dataset. We also sample equidistant points on the language-aligned path to come up with LAW#2, LAW#4 and LAW#15 containing two, four and fifteen waypoints, respectively. The intuition is that LAW `pano` takes the agent back to the language-aligned path some distance ahead of its position, while LAW `step` brings it directly to the path.

Quantitative Results. In Table 1, we see that LAW `pano`, supervised with the language-aligned path performs better than the base model supervised with the goal-oriented path, across all metrics in both validation seen and unseen environments. We observe the same trend in the Waypoint Accuracy (WA) metric that we introduced. Table 2 shows that agents perform similarly even when we vary the number of waypoints for the language-aligned path supervision, since all of them essentially follow the path. This could be due to relatively short trajectory length in the R2R dataset (average of 10m) making LAW `pano` denser than needed for the instructions. To check this, we analyze the sub-instruction data and find that one sub-instruction (e.g. 'Climb up the stairs') often maps to several `pano` waypoints, suggesting fewer waypoints are sufficient to specify the language-aligned path. For such paths, we find that the LAW #4 is better than the LAW `pano` (see supplement for details). Figure 3 further analyzes model performance by grouping episodes based on similarity between the goal-oriented shortest path and the language-aligned path in the ground truth trajectories (measured by nDTW). We find that the LAW model performs better than the goal-oriented model, especially on episodes with dissimilar paths (lower nDTW) across both the nDTW and Waypoint Accuracy metrics.

Qualitative Analysis. To interpret model performance concretely with respect to path alignment we use the FG-R2R data, which contains mapping between sub-instructions and waypoints. Figure 4

[3] https://github.com/jacobkrantz/VLN-CE

Training	Val-Seen					Val-Unseen				
	SR↑	SPL↑	nDTW↑	sDTW↑	WA↑	SR↑	SPL↑	nDTW↑	sDTW↑	WA↑
`goal`	0.34	0.32	0.54	0.29	0.48	0.29	0.27	0.50	0.24	0.41
LAW `pano`	**0.40**	**0.37**	**0.58**	**0.35**	**0.56**	**0.35**	**0.31**	**0.54**	**0.29**	**0.47**

Table 1: **Goal only vs language-aligned waypoint (LAW) supervision.** LAW `pano` performs better than `goal` across all metrics, including the instruction-following metrics, nDTW and Waypoint Accuracy. This suggests that language-aligned supervision encourages the agent to follow instructions better than goal-oriented supervision.

LAW	Distance between waypoints	Val-Seen				Val-Unseen			
		SR↑	SPL↑	nDTW↑	sDTW↑	SR↑	SPL↑	nDTW↑	sDTW↑
#2	5.00m	0.39	0.36	0.57	0.34	0.33	0.30	0.52	0.28
#4	2.50m	0.35	0.33	0.54	0.30	0.34	0.31	0.53	0.29
`pano` (6)	2.00m	**0.40**	**0.37**	**0.58**	**0.35**	**0.35**	**0.31**	**0.54**	**0.29**
#15	0.60m	0.34	0.32	0.54	0.29	0.33	0.30	0.52	0.28
`step`	0.25m	0.37	0.35	0.57	0.32	0.32	0.30	0.53	0.27

Table 2: **Varying density of language-aligned supervision from very sparse (#2) to dense (`step`).** This study shows that with varying density of the language-aligned waypoint supervision, the agent performs similarly, since all of them essentially follow the same path.

contrast the agent trajectories of the LAW `pano` and goal-oriented agents on an unseen scene. We observe that the path taken by the LAW agent conforms more closely to the instructions (also indicated by higher nDTW). We present more examples in the supplement.

Additional Experiments. We additionally experiment with mixing goal-oriented and language-oriented losses while training, but observe that they fail to outperform the LAW `pano` model. The best performing mixture model achieves 53% nDTW in unseen environment, as compared to 54% nDTW for LAW `pano` (see supplement). Moreover, we perform a set of experiments on the recently introduced VLN-CE RxR dataset and observe that language-aligned supervision is better than goal-oriented supervision for this dataset as well, with LAW `step` showing a 6% increase in WA and 2% increase in nDTW over `goal` on the unseen environment. We defer the implementation details and results to the supplement.

5 Conclusion

We show that instruction following during the VLN task can be improved using language-aligned supervision instead of goal-oriented supervision as commonly employed in prior work. Our quantitative and qualitative results demonstrate the benefit of the LAW supervision. The waypoint accuracy metric we introduce also makes it easier to interpret how agent navigation corresponds to following sub-instructions in the input natural language. We believe that our results show that LAW is a simple but useful strategy to improving VLN-CE.

Acknowledgements We thank Jacob Krantz for the VLN-CE code on which this project was based, Erik Wijmans for initial guidance with reproducing the original VLN-CE results, and Manolis Savva for discussions and feedback. We also thank the anonymous reviewers for their suggestions and feedback. This work was funded in part by a Canada CIFAR AI Chair and NSERC Discovery Grant, and enabled in part by support provided by WestGrid and Compute Canada.

References

Peter Anderson, Angel Chang, Devendra Singh Chaplot, Alexey Dosovitskiy, Saurabh Gupta, Vladlen Koltun, Jana Kosecka, Jitendra Malik, Roozbeh Mottaghi, and Manolis Savva. 2018a. On evaluation of embodied navigation agents. arXiv preprint arXiv:1807.06757.

Peter Anderson, Ayush Shrivastava, Devi Parikh, Dhruv Batra, and Stefan Lee. 2019. Chasing ghosts: Instruction following as bayesian state tracking. In Advances in neural information processing systems.

Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. 2018b. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In Proc. of Conference on Computer Vision and Pattern Recognition (CVPR).

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. 2017. Matterport3D: Learning from RGB-D data in indoor environments. In Proc. of International Conference on 3D Vision (3DV).

Howard Chen, Alane Suhr, Dipendra Misra, Noah Snavely, and Yoav Artzi. 2019. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In Proc. of Conference on Computer Vision and Pattern Recognition (CVPR).

Kevin Chen, Junshen K Chen, Jo Chuang, Marynel Vázquez, and Silvio Savarese. 2021. Topological Planning with Transformers for Vision-and-Language Navigation. In Proc. of Conference on Computer Vision and Pattern Recognition (CVPR).

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In Proc. of Conference on Computer Vision and Pattern Recognition (CVPR).

Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. 2018. Speaker-follower models for vision-and-language navigation. In Advances in neural information processing systems.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In Proc. of Conference on Computer Vision and Pattern Recognition (CVPR).

Yicong Hong, Cristian Rodriguez-Opazo, Qi Wu, and Stephen Gould. 2020. Sub-instruction aware vision-and-language navigation. In Proc. of the conference on empirical methods in natural language processing (EMNLP).

Ronghang Hu, Daniel Fried, Anna Rohrbach, Dan Klein, Trevor Darrell, and Kate Saenko. 2019. Are You Looking? Grounding to Multiple Modalities in Vision-and-Language Navigation. In Proc. of the Conference of the Association for Computational Linguistics (ACL).

Gabriel Ilharco, Vihan Jain, Alexander Ku, Eugene Ie, and Jason Baldridge. 2019. General evaluation for instruction conditioned navigation using dynamic time warping. arXiv preprint arXiv:1907.05446.

Vihan Jain, Gabriel Magalhaes, Alexander Ku, Ashish Vaswani, Eugene Ie, and Jason Baldridge. 2019. Stay on the path: Instruction fidelity in vision-and-language navigation. In Proc. of the Conference of the Association for Computational Linguistics (ACL).

Jacob Krantz, Aaron Gokaslan, Dhruv Batra, Stefan Lee, and Oleksandr Maksymets. 2021. Waypoint Models for Instruction-guided Navigation in Continuous Environments. In Proc. of International Conference on Computer Vision (ICCV).

Jacob Krantz, Erik Wijmans, Arjun Majumdar, Dhruv Batra, and Stefan Lee. 2020. Beyond the Nav-Graph: Vision-and-Language Navigation in Continuous Environments. In Proc. of European Conference on Computer Vision (ECCV).

Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. 2020. Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding. In Proc. of the conference on empirical methods in natural language processing (EMNLP).

Chih-Yao Ma, Jiasen Lu, Zuxuan Wu, Ghassan AlRegib, Zsolt Kira, Richard Socher, and Caiming Xiong. 2019. Self-monitoring navigation agent via auxiliary progress estimation. In Proc. of International Conference on Learning Representations (ICLR).

Harsh Mehta, Yoav Artzi, Jason Baldridge, Eugene Ie, and Piotr Mirowski. 2020. Retouchdown: Adding Touchdown to StreetLearn as a shareable resource for language grounding tasks in street view. In Proc. of the Third International Workshop on Spatial Language Understanding.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, and Luca Antiga. 2019. Pytorch: An imperative style, high-performance deep learning library. In Advances in neural information processing systems.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In Proc. of the conference on empirical methods in natural language processing (EMNLP).

Yuankai Qi, Zizheng Pan, Shengping Zhang, Anton van den Hengel, and Qi Wu. 2020. Object-and-action aware model for visual language navigation. In Proc. of European Conference on Computer Vision (ECCV).

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. 2011. A reduction of imitation learning and structured prediction to no-regret online learning. In Proc. of the International Conference on Artificial Intelligence and Statistics (AISTATS).

Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, and Jitendra Malik. 2019. Habitat: A platform for embodied AI research. In Proc. of International Conference on Computer Vision (ICCV).

Hao Tan, Licheng Yu, and Mohit Bansal. 2019. Learning to navigate unseen environments: Back translation with environmental dropout. In Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics (NAACL).

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Advances in neural information processing systems, volume 30, pages 5998–6008.

Xin Wang, Qiuyuan Huang, Asli Celikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. 2019. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. In Proc. of Conference on Computer Vision and Pattern Recognition (CVPR).

Erik Wijmans, Abhishek Kadian, Ari Morcos, Stefan Lee, Irfan Essa, Devi Parikh, Manolis Savva, and Dhruv Batra. 2020. DD-PPO: Learning near-perfect pointgoal navigators from 2.5 billion frames. In Proc. of International Conference on Learning Representations (ICLR).

Wang Zhu, Hexiang Hu, Jiacheng Chen, Zhiwei Deng, Vihan Jain, Eugene Ie, and Fei Sha. 2020. BabyWalk: Going farther in vision-and-language navigation by taking baby steps. In Proc. of the Conference of the Association for Computational Linguistics (ACL).

A Appendix

A.1 Glossary

Some commonly used terminologies in this work are described here:

- LAW refers to language-aligned waypoints such that the navigation path aligns with the language instruction.
- nav-graph refers to the discrete navigation graph of a scene.
- `pano` refers to the reference paths constructed by taking the discrete nav-graph nodes corresponding to positions of panoramic cameras in the R2R dataset.
- `step` refers to the reference paths constructed by taking the shortest geodesic distance between the `pano` paths to create dense waypoints corresponding to an agent step size of $0.25m$.
- 'shortest' refers to the goal-oriented path, i.e, the shortest path to the goal.
- `goal` refers to the model supervised with only the goal.

A.2 Analysis of VLN-CE R2R path

We analyze the similarity of the VLN-CE R2R reference path to the shortest path using nDTW. We find that $\sim 6\%$ of episodes (including training and validation splits), have nDTW(shortest, LAW `step`) < 0.8.

Figure 5 shows the distribution of nDTW of the ground truth trajectories (LAW `step`) against the shortest path (goal-oriented action sensor) and LAW `pano` (language-aligned action sensor). It shows that the two distributions are different and that the language-aligned sensor will be much closer to the ground truth trajectories. Figure 6 shows percentage of unseen episodes binned by nDTW value of reference path to shortest path, which helps us analyze our model performance as shown in Figure 3 (main paper). Additionally, we visualize a few such paths to see how dissimilar they are in Figure 7.

A.3 CMA Model

The Cross-Modal Attention (CMA) Model takes the input RGB and depth observations and encode them using a ResNet50 (He et al., 2016) pre-trained on ImageNet (Deng et al., 2009) and a modified ResNet50 trained on point-goal navigation (Wijmans et al., 2020) respectively. It also takes as input the GLoVE (Pennington et al., 2014) embeddings for the tokenized words in the language instruction and pass them through a bi-directional LSTM to obtain their feature representations. The CMA model consists of two recurrent (GRU) networks. The first GRU encodes a history of the agent state, which is then used to generate attended instruction features. These attended instruction features are in turn used to generate visual attentions. The second GRU takes in all the features generated thus far to predict an action. The attention used here is a scaled dot-product attention (Vaswani et al., 2017).

A.4 Full Evaluation

A.4.1 Metrics

We report the full evaluation of the models here on the standard metrics for VLN such as:
Trajectory Length (TL): agent trajectory length.
Navigation Error (NE): distance from agent to goal at episode termination.
Success Rate (SR): rate of agent stopping within a threshold distance (around 3 meters) of the goal.
Oracle Success Rate (OS): rate of agent reaching within a threshold distance (around 3 meters) of the goal at any point during navigation.
Success weighted by inverse Path Length (SPL): success weighted by trajectory length relative to shortest path trajectory between start and goal.
Normalized dynamic-time warping (nDTW): evaluates how well the agent trajectory matches the ground truth trajectory.
Success weighted by nDTW (SDTW): nDTW, but calculated only for successful episodes.

A.4.2 Quantitative Results

We observe that the models in the ablation (LAW #2 to LAW `step` in Table 3) perform similarly, which could be due to the fact that the average trajectory length in the R2R dataset is around 10m and the LAW `pano` is actually denser than the agent needs to follow instructions. We analyze this by using the sub-instruction data and find that one sub-instruction often maps to several `pano` waypoints and the language-aligned path can be explained via fewer waypoints. We show some such examples from the dataset in Figure 8. We also report the results on the R2R test split in Table 4, which shows that LAW `pano` performs better on OS, while performing similarly to `goal` on SR and SPL metrics.

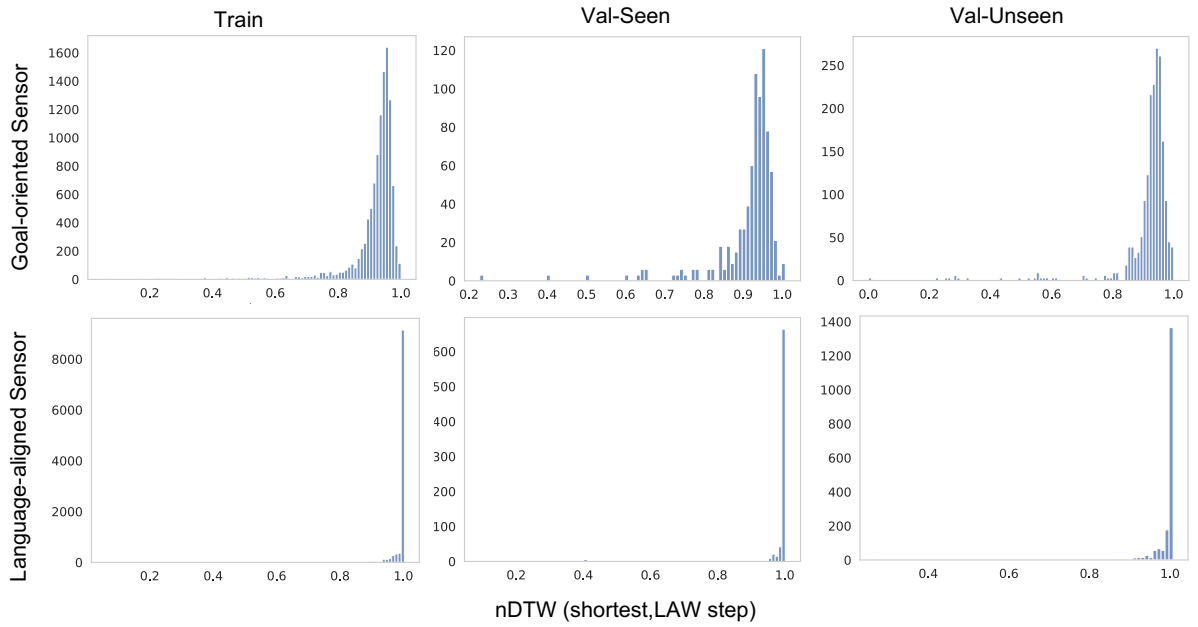

Figure 5: Plots showing a distribution of the number of R2R episodes across different nDTW values of reference path to shortest path for train, val-seen and val-unseen splits. There are many episodes for which the goal-oriented shortest path does not match the language-aligned path, as generated by the goal-oriented action sensor (top). We mitigate this problem by using language-aligned action sensor (bottom).

LAW	Distance between waypoints	Val-Seen								Val-Unseen							
		TL↓	NE↓	OS↑	SR↑	SPL↑	nDTW↑	sDTW↑	WA↑	TL↓	NE↓	OS↑	SR↑	SPL↑	nDTW↑	sDTW↑	WA↑
#1(goal)	10m	9.06	7.21	0.44	0.34	0.32	0.54	0.29	0.48	8.27	7.60	0.36	0.29	0.27	0.50	0.24	0.41
#2	5m	9.39	6.76	0.47	0.39	0.36	0.57	0.34	0.51	8.57	7.41	0.39	0.33	0.30	0.52	0.28	0.44
#4	2.5m	9.08	6.94	0.47	0.35	0.33	0.54	0.30	0.49	8.57	7.01	0.41	0.34	0.31	0.53	0.29	0.44
pano	2m	9.34	6.35	0.49	**0.40**	**0.37**	**0.58**	**0.35**	**0.56**	8.89	6.83	0.44	**0.35**	**0.31**	**0.54**	**0.29**	**0.47**
#15	0.6m	9.51	7.16	0.45	0.34	0.32	0.54	0.29	0.50	8.71	7.05	0.41	0.33	0.30	0.52	0.28	0.44
step	0.25m	9.76	6.35	0.49	0.37	0.35	0.57	0.32	0.50	9.06	6.81	0.40	0.32	0.30	0.53	0.27	0.44

Table 3: LAW pano model supervised with language-aligned waypoints performs better than the same model supervised with goal-oriented path, i.e. the shortest path to the goal. All models supervised with language-aligned path, but with varying density, perform similarly.

Training	Test				
	TL↓	NE↓	OS↑	SR↑	SPL↑
goal	8.85	7.91	0.36	0.28	0.25
LAW pano	9.67	7.69	**0.38**	0.28	0.25

Table 4: Evaluating LAW pano on the VLN-CE test split gives us an increase in OS, although the SR and SPL is same as the goal.

However, since the VLN-CE leaderboard[4] does not report the instruction-following metrics, nDTW and sDTW, we could not report how well the LAW pano agent follows instructions on the test set.

[4]https://eval.ai/web/challenges/challenge-page/719

A.4.3 Qualitative Analysis

Table 5 shows a qualitative interpretation of some R2R unseen episodes for the two models goal and LAW pano, along with the sub-instruction data from the FG-R2R dataset. We see that LAW pano is able to get more number of waypoints (and hence sub-instructions) correct than the goal model. We report the Waypoint Accuracy metric at threshold distances of 0.5m and 1.0m for the same. It also shows that Waypoint Accuracy is more intuitive than nDTW in terms of interpreting what fraction of waypoints the agent is able to predict correctly.

A.5 Mixing goal-oriented and language-oriented losses

We experiment with mixing goal-oriented loss (G) and language-oriented loss (L) during training

CMA + goal	CMA + LAW pano
nDTW=0.52, WA@0.5m=0.4, WA@1.0m=0.8 1→4 Walk straight through the living room towards the stairs. 4→5 Go to the right of the stairs towards the dining area 5→5 and wait by the leather chair at the entry to the dining room.	nDTW=0.98, WA@0.5m=1.0, WA@1.0m=1.0 1→4 Walk straight through the living room towards the stairs. 4→5 Go to the right of the stairs towards the dining area 5→5 and wait by the leather chair at the entry to the dining room.
nDTW=0.15, WA@0.5m=0.4, WA@1.0m=0.6 1→3 Walk straight into the kitchen area. 3→4 Turn left and exit the kitchen 4→5 and stop there.	nDTW=0.74, WA@0.5m=0.8, WA@1.0m=1.0 1→3 Walk straight into the kitchen area. 3→4 Turn left and exit the kitchen 4→5 and stop there.
nDTW=0.26, WA@0.5m=0.2, WA@1.0m=0.2 1→4 Go to exit of writing room. (1 correct) 4→5 Stop between pillars.	nDTW=0.95, WA@0.5m=1.0, WA@1.0m=1.0 1→4 Go to exit of writing room. 4→5 Stop between pillars.

Table 5: We analyse model performance on sub-instruction data (2020). CMA + LAW pano (right) correctly predicts more sub-instructions compared to CMA + goal (left). Mapping between sub-instruction and waypoints is indicated by start and end waypoint indices. Green and Red indicate correct and incorrect prediction respectively. WA@0.5m and WA@1.0m indicate Waypoint Accuracy measured at a threshold distance of 0.5m and 1.0m respectively from the waypoint.

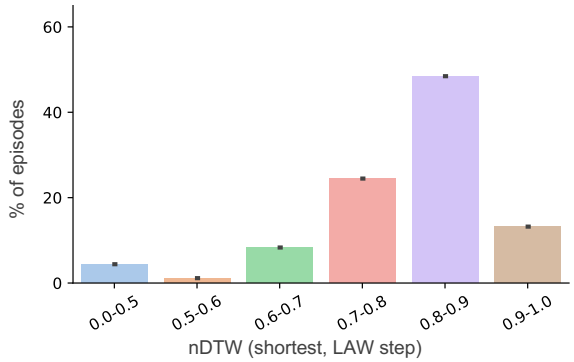

Figure 6: Plot showing percentage of episodes in R2R unseen dataset binned by nDTW value of reference path to shortest path. The lower nDTW values indicate episodes for which the goal-oriented shortest path does not match language-aligned path.

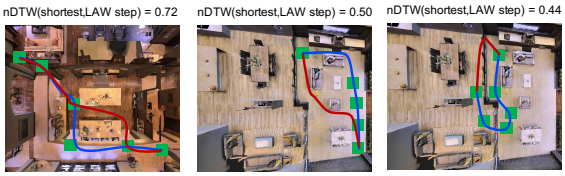

Figure 7: Visualization of a few R2R unseen episodes with nDTW(shortest,LAW step)< 0.8 shows us how dissimilar the goal-oriented shortest path (red) and the language-aligned path (blue) are.

to further understand the contribution language-oriented supervision. We pre-trained with G using teacher forcing and then fine-tuned with (a) only L, (b) L+G, (c) randomly chosen L or G, using DAgger. The results as reported in Table 6 show that none of the models outperform LAW pano, indicating that training with mixed losses fail to perform as well as training with only language-oriented loss.

A.6 Evaluation on VLN-CE RxR dataset

Dataset. Beyond the R2R dataset, there exist VLN datasets where the aim is to have the language-aligned path not be the shortest path. Jain et al. (2019) proposed the Room-for-Room (R4R) dataset by combining paths from R2R. More recently, Ku et al. (2020) introduced the Room-Across-Room (RxR) dataset that consists of new trajectories designed to not match the shortest path between start and goal. Importantly, they do not have a bias on the path length itself. The RxR dataset is 10x larger than R2R and consists of longer trajectories and instructions in three languages, English, Hindi, and Telugu. Both the R4R and RxR datasets are on the discrete nav-graph setting. However, the RxR dataset has been recently ported to the continuous state space of the VLN-CE for the RxR-Habitat challenge at CVPR 2021[5]. We experiment on the VLN-CE RxR to further investigate if language-aligned supervision is better than goal-oriented supervision on a dataset other than R2R.

Model. We build our experiments on the model architecture provided for the VLN-CE RxR Habitat challenge. The only difference in the CMA architecture in VLN-CE RxR from the one used in VLN-CE is that they use pre-computed BERT features for the language instructions instead of GLoVE

[5] https://github.com/jacobkrantz/VLN-CE/tree/rxr-habitat-challenge

Training	Val-Seen						Val-Unseen					
	OS↑	SR↑	SPL↑	nDTW↑	sDTW↑	WA↑	OS↑	SR↑	SPL↑	nDTW↑	sDTW↑	WA↑
LAW `pano`	**0.49**	**0.40**	**0.37**	**0.58**	**0.35**	**0.56**	**0.44**	**0.35**	**0.31**	**0.54**	**0.29**	**0.47**
G→L	0.49	0.36	0.34	0.56	0.32	0.56	0.40	0.32	0.29	0.51	0.27	0.47
G→G+L	0.46	0.34	0.31	0.57	0.30	0.55	0.38	0.27	0.25	0.51	0.23	0.45
G→G-or-L	0.45	0.35	0.34	0.55	0.31	0.51	0.41	0.32	0.29	0.53	0.27	0.46

Table 6: Experiments show that models trained with a mixture of goal-oriented (G) and language-oriented (L) superivision underperforms the model trained with only our language-oriented loss.

Training	Val-Seen								Val-Unseen							
	TL↓	NE↓	OS↑	SR↑	SPL↑	nDTW↑	sDTW↑	WA↑	TL↓	NE↓	OS↑	SR↑	SPL↑	nDTW↑	sDTW↑	WA↑
`goal`	4.55	12.01	0.14	0.06	0.05	0.34	0.05	0.28	4.03	11.00	0.16	0.06	0.05	0.36	0.05	0.27
LAW `pano`	6.27	12.07	0.17	**0.09**	**0.09**	0.35	0.08	0.31	4.62	11.04	0.16	**0.10**	**0.09**	0.37	0.08	0.28
LAW `step`	7.92	11.94	0.20	0.07	0.06	**0.36**	0.06	**0.35**	4.01	10.87	0.21	0.08	0.08	**0.38**	0.07	**0.33**

Table 7: Experiments on the recently released RxR-Habitat benchmark (English language split) show that LAW methods outperform the `goal`, with LAW `step` having a 6% increase in WA and 2% increase in nDTW over `goal` on unseen environment. This indicates that our idea of language-aligned supervision is useful beyond R2R.

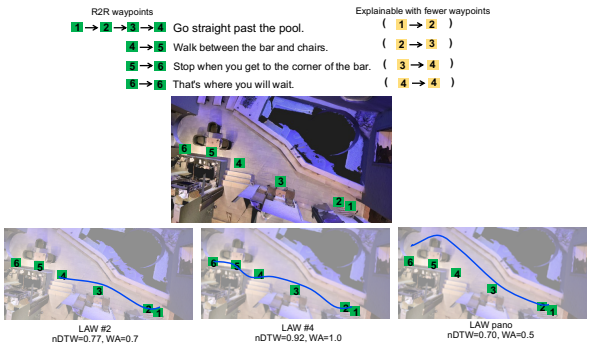

Figure 8: Top: Analysis of the R2R dataset along with the FG-R2R shows that one sub-instruction often maps to several `pano` waypoints and can be explained via fewer waypoints.
Bottom: The above path can be defined via 4 waypoints and evaluating the episode with the different model variations shows that LAW #4 (supervising with 4 waypoints) performs the best.

embeddings. We vary the nature of supervision as before. The `goal` model receives goal-oriented supervision, whereas the LAW `pano` and LAW `step` are supervised with language-aligned `pano` and `step` waypoints, respectively. The baseline model in the VLN-CE RxR codebase also follows the `step` supervision.

Training. We train the methods with teacher forcing as was done in the baseline model in the VLN-CE RxR Habitat challenge. However, we used only the RxR English language split for both training and evaluating our models. Note that the training regime here is different from that of our main R2R experiments and does not have the DAgger fine-tuning phase.

Results. Table 7 shows the results of our experiments. We observe that the LAW methods outperform the `goal`, with LAW `step` showing a 6% increase in WA and 2% increase in nDTW on the unseen environment. This indicates that language-aligned supervision is useful beyond the R2R dataset.