Multi-modal Intent Classification for Assistive Robots with Large-scale Naturalistic Datasets

Karun Mathew [♣0] **Venkata S Aditya Tarigoppula** [♦♥] **Lea Frermann** [♠]

[♣]Newline Structures Pvt Ltd.

[♦]Department of Biomedical Engineering, The University of Melbourne

[♥]ARC Training Centre in Cognitive Computing for Medical Technologies

[♠]School of Computing and Information Systems, The University of Melbourne

karunmatthew@live.in aditya.tarigoppula@gmail.com lfrermann@unimelb.edu.au

Abstract

Recent years have brought a tremendous growth in assistive robots/prosthetics for people with partial or complete loss of upper limb control. These technologies aim to help the users with various reaching and grasping tasks in their daily lives such as picking up an object and transporting it to a desired location; and their utility critically depends on the ease and effectiveness of communication between the user and robot. One of the natural ways of communicating with assistive technologies is through verbal instructions. The meaning of natural language commands depends on the current configuration of the surrounding environment and needs to be interpreted in this multi-modal context, as accurate interpretation of the command is essential for a successful execution of the user's intent by an assistive device. The research presented in this paper demonstrates how large-scale situated natural language datasets can support the development of robust assistive technologies. We leveraged a navigational dataset comprising > 25k human-provided natural language commands covering diverse situations. We demonstrated a way to extend the dataset in a task-informed way and use it to develop multi-modal intent classifiers for pick and place tasks. Our best classifier reached > 98% accuracy in a 16-way multi-modal intent classification task, suggesting high robustness and flexibility.

1 Introduction

Paralysis is a loss of motor function to varying degrees of severity often resulting in severely reduced or complete loss of upper and/or lower limb control. Such impairments reduce the quality of life for millions of people affected by paralysis (Armour et al., 2016) and increase their dependence upon others to perform day-to-day activities including self- or object locomotion and object manipulation tasks like reaching, picking up an object and moving it to a desired location (pick and place). Assistive devices can compensate for some of the impairments provided that they can accurately infer and execute user intents. Most assistive devices currently in use rely on manual control (e.g., wheelchairs controlled with joysticks), and cannot understand natural language user commands or map them to potentially complex sequences of actions. Moreover, they do not perceptively account for the surrounding environment they are interacting with and as a consequence require a more detailed user input. Therefore, recent developments have focused on Intelligent Assistive Devices (IAD), that combine traditional assistive devices with advanced sensors and artificial intelligence, aiming for an accurate inference of a user's intent in the context of a multi-modal representation of the environment (Barry et al., 1994).

The utility of the IAD depends critically on the efficiency and effectiveness of the communication

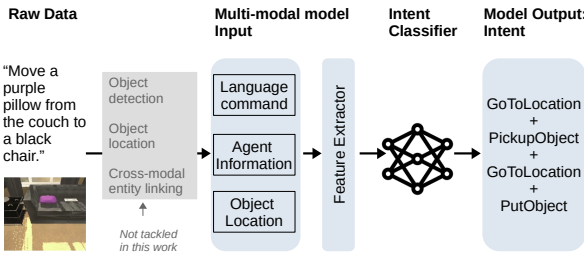

Figure 1: High-level overview of our intent classifier. The system receives visual information extracted from the environment together with a natural language task command as input; and uses this to predicts the *intent* as a suitable sequence of actions necessary to execute the command. Visual scene parsing and cross-modal entity linking are not tackled in this work.

[0]Work done while at Melbourne University.

with the user. One of the natural ways of instructing the IAD is through verbal communication. It is important to recognize that a majority of patients suffering a loss of limb control retain the ability to speak, albeit impaired in some cases. Modern voice controlled IADs such as wheelchairs (Hou et al., 2020; Umchid et al., 2018), smart home appliances and assistive anthropomorphic robots (Pulikottil et al., 2018a; John et al., 2020) are still limited to a pre-defined set of instructions that the user can choose from. This requires the user to explicitly dictate each individual action leading to the final goal rather than just stating the desired goal alone and off-loading the decision making to perform any required sequence of actions to accomplish the user's intent. Consider the example in Figure 1, where a robotic assistant situated in a complex and dynamic environment is given a verbal instruction "Pick up the book". While the need of a "pick" action is evident from the language command alone, possible additional actions (navigate to the book's location, or to turn around to face the book) depend on the agent and book's location, thus requiring an interpretation of the natural language command in the context of the surrounding environment.

In this paper, we present a step towards bridging this gap by drawing on large, data resources, state of the art language understanding and intent classification methods. We develop a classifier that takes a higher-order task command contextualized in the current environment as input and derives the necessary set of sub-actions (intents) required to achieve the goal intended by the user. We present a scalable framework to develop such flexible natural language interfaces for IAD that execute 'pick and place' tasks. Specifically, we leverage ALFRED (Shridhar et al., 2020), a large-scale naturalistic data set for developing indoor navigation systems, comprising diverse, crowd-sourced natural language commands and photo-realistic images, and adapt it to the pick-and-place task (Section 3). We augment the state-of-the-art natural language intent classifier DIET (Bunk et al., 2020) with a visual processing component (Section 4). Evaluation against simpler classifiers as well as exclusively text-based classification scenarios shows the advantage of joint processing of visual and language information, as well as the DIET architecture. The use of large-scale naturalistic data allows to build solutions that generalize beyond the confines of a laboratory, are easily adaptable and have the potential to improve the overall quality of life for the user. This framework is part of a larger project intended to develop a multi-modal (voice and brain signal) prosthetic limb control.

In short, our contributions are:

- We show that task-related large-scale data sets can effectively support the development assistive technology. We augmented the ALFRED data set with anticipated scenarios of human-assistive agent interaction, including noisy and partially observed scenarios.
- We contribute a multi-modal extension of a state-of-the-art natural language intent classifier (DIET) with a visual component, which lead to the overall best classification results.
- Our best performing model achieved 98% accuracy in a 16-way classification task over diverse user-generated commands, evidencing that our architecture supports flexible and reliable intent classification.

2 Related Work

Our work is cross-disciplinary, covering both medical robotics and (multi-modal) machine learning and NLP for intent classification.

Intent classification is the task of mapping a natural language input to a set of actions that when executed help achieve the underlying goals of the user. As an essential component of conversational systems, it has attracted much attention in the natural language understanding community with methods ranging from semantic parsing (Chen and Mooney, 2011) to more recent deep learning (Liu and Lane, 2016; Goo et al., 2018) and transfer learning approaches, with unsupervised pre-training (Liu et al., 2019; Henderson et al., 2020). The on-line interactive nature of dialogue applications makes model efficiency a central objective. We build on the recent DIET classifier (Dual Intent and Entity Transformer; (Bunk et al., 2020)) which achieves competitive performance in intent classification, while maintaining a lightweight architecture without the need for a large pre-trained language models. DIET was originally developed for language-based dialogue, and we extend the system with a vision understanding component and show that it generalizes to a multi-modal task setup.

Visually grounded language understanding addresses the analysis of verbal commands in the context of the visual environment. Prior work ranges from schematic representations of the environment avoiding the need for image analysis (Chen and Mooney, 2011) over simplistic visual environments ("block worlds" Bisk et al. (2016)) to complex outdoor navigation (Chen et al., 2019)). The advance of deep learning methods for joint visual and textual processing has lead to the development of large-scale datasets which feature both naturalisitic language as well as images (Bunk et al., 2020; Chen et al., 2019; Puig et al., 2018). We leverage a subset of the ALFRED dataset (Bunk et al., 2020) which is a benchmark dataset for learning a mapping from natural language instructions and egocentric (first person) vision to sequences of actions for performing household tasks. The commands in the ALFRED dataset are crowd-sourced from humans, and as such are diverse and resemble naturalistic language. The visual scenes are complex and photo-realistic, and the dataset contains tasks requiring the agent to execute complex sequences of multiple, context-dependent actions to manipulate objects in an environment that closely resembles the medical application scenario addressed in this paper. We note that we do not address the object recognition challenge in this work, but assume access to the object locations, and train intent classifiers to incorporate such information.

Interfacing medical assistive technologies Traditional interfaces to assistive technologies involved manipulating joysticks (House et al., 2009), or verbal commands which are restricted to simple templates. The latter include very simple templates ("up", "down", "left"; Pulikottil et al. (2018b)), or highly constrained training data sets based on command templates produced by five human annotators (Stepputtis et al., 2020). In this paper, we leverage natural commands produced by thousands of crowd workers with the aim to produce a robust intent classifier amenable to natural speech input.

3 Data

We leveraged and extended the ALFRED (Action Learning From Realistic Environments and Directives) dataset of visually grounded language commands (Shridhar et al., 2020), for training and testing our intent classifier. ALFRED consists of more than 8,000 sets of scenes with unique environmental layout with a fixed set of associated movable and static objects. Each scene is paired with an indoor navigation task, and contains three levels of information: (1) positional information of the agent and objects, (2) natural language descriptions of the high-level task and low-level instructions to achieve the goal, and (3) a sequence of discrete actions to be performed by the agent to achieve the goal. An example is shown in Figure 2.

The visual task information comprises the positional (x, y, z) co-ordinates of the agent (Agent Information), and the positional information of static and interactable objects in the environment (Scene Information). The natural language annotation includes a "high-level task" describing the overall goal, as well as detailed low-level instructions ("low-level subtasks") on how to achieve the goal. Low level instructions were provided by at least three human annotators through crowdsourcing. Finally, each ALFRED task in the train and validation set is augmented with an "action plan" listing the sequence of actions (or intents) such as `GoToLocation` or `PickUpObject` required to achieve the goal in the context of the scene configuration (Figure 2, bottom). Crowd workers were prompted by these action plans, so that a gold-standard utterance-intent alignment could be derived from the data set.

3.1 ALFRED for intent classification

We utilized a subset of the dataset corresponding to "pick and place" tasks, which is most relevant to our target application of humanoid arm control. We refer to the item that is to be picked up as "target object" and the item on which the picked-up object is to be placed as the "receptacle object". ALFRED contains around 3,000 different pick and place tasks, involving 58 unique target objects and 26 receptacle objects across 120 indoor scenes.

Leveraging the ALFRED action plans, we could map all "pick and place" language commands to a combination of three unique sub-actions: `GoToLocation`[1], `PickUpObject` and `PutObject`. `GoToLocation` actions referred to actions of the agent moving to a given location. `PickUpObject` and `PutObject` corresponded to the action of picking up the target object and placing the target object, respectively. Note that a single natural language directive can cover one or more atomic actions. We refer to com-

[1] in analogy to a lateral or vertical movement of the robotic arm

Agent Information	Agent	{x: -2.50, y: 0.92, z: 2.50, rotation=0}
Scene Information	FloorPlan:	FloorPlan214
	Plate,	{x: -0.31, y: 0.27, z: 5.99}
	WateringCan,	{x: -2.28, y: 0.45, z: 4.27}
	KeyChain,	{x: -4.31, y: 0.45, z: 6.73}
	Box,	{x: -2.40, y: 0.57, z: 4.57}
	Laptop,	{x: -2.49, y: 0.53, z: 0.79}
	Vase,	{x: -0.60, y: 1.46, z: 5.74}
	WateringCan,	{x: -2.40, y: 0.44, z: 3.83}
Language Information	High Level Task	"Move the purple pillow from the couch to the black chair."
	Low Level Subtask 1	"Turn right and walk up to the couch."
	Low Level Subtask 2	"Pick up the purple pillow off of the couch."
	Low Level Subtask 3	"Turn around and walk across the room, then hand a left and walk over to the black chair."
	Low Level Subtask 4	"Put the purple pillow on the black chair."
Action Plan	Discrete Action 1	`GoToLocation`
	Discrete Action 2	`PickUpObject`
	Discrete Action 3	`GoToLocation`
	Discrete Action 4	`PutObject`

Figure 2: Visual and Language information corresponding to a pick and place task in ALFRED, as well as the associated Action Plan, i.e., sequence of actions (or intents), as provided in the the data set.

mands describing a single task as "single intent" (*"Pick up the keys."*), and commands describing multiple tasks as "multi-intent" (*"Bring the keys from the chair to the table.""*). Table 1 illustrates the range of tasks and intents supported by the original ALFRED dataset and resulting training instances. In the original ALFRED data set, each low-level instruction was associated with a single intent (Table 1 middle).

We augmented high-level task descriptions with intents by concatenating the actions of its associated low-level tasks (Table1 top). In addition, we augmented the ALFRED tasks with additional diverse and relevant scenarios to our assistive agent use case. First, we created partial tasks where the agent was required to execute only parts of the complete pick and place action sequence (e.g., only move to, and pick up the object). We synthesized these instances by concatenating all possible ordered subsequences of the low- level sub-tasks for a scenario and concatenating their corresponding natural language commands. The resulting instances were then treated as a single "multi-intent" directive (Table 1, bottom). Second, we randomized the positions of the target and receptacle objects mentioned in the verbal commands to (1) far from the agent, (2) near the agent or (3) near the receptacle.

Finally, we imposed physical constraints onto the agent, resembling the characteristics of an assistive robotic arm. In the original ALFRED, all objects within a specific distance of the agent are considered 'pickable'. We introduced a threshold (60 degrees) beyond which an object is unreachable and requires the agent to turn to the object first. We introduced a corresponding new action called `RotateAgent` that needed to be performed before the `PickUpObject`. In addition, we reduced the maximum reach distance of the agent to 0.5 meters and updated ALFRED tasks accordingly with `GoToLocation` actions before `PickUpObject` where necessary. The resulting dataset more realistically represented the physical constraints faced by real world entities, and the actions to be taken to meet the necessary preconditions to perform a task. We also handled cases where visual features corresponding to a language command were missing or irrelevant. For example, the command "Take a step forward", has a single intent `GoToLocation` when considering the natural language command alone. For such commands, we generated multiple data instances with randomized visual features to encourage the model to be insensitive to an irrelevant input modality.

We divided our final dataset into non-overlapping training, testing and validation sets with no overlap in environments. We treated each unique action combination observed in the data as a distinct intent, leading to a total of 16 possible intents that could be selected in response to a spoken command.[2] Table 2 summarizes our data set, full

[2]In addition to the 9 unique intents in Table 1, these are {`PickUpObject`, `PutObject`}, {`RotateAgent`,

Intent type		Command	Intent
High-level single intent	1.	*"Move a red pillow from the couch to a black chair."*	{`GoToLocation`, `PickUpObject`, `GoToLocation`, `PutObject`}
Low-level single intent	2.	*"Turn right and walk up to the couch."*	{`GoToLocation`}
	3.	*"Pick up the red pillow off the couch."*	{`PickUpObject`}
	4.	*"Turn around and walk... to the chair."*	{`GoToLocation`}
	5.	*"Put the red pillow on the chair."*	{`PutObject`}
Low-level multi intent	6.	*"Turn right and walk up to the couch. Pick up the red pillow off the couch."*	{`GoToLocation`, `PickUpObject`}
	7.	*"Pick up the red pillow off the couch. Turn around and walk ... to the chair."*	{`PickUpObject`, `GoToLocation`}
	8.	*"Turn around and walk ... to the chair. Put the red pillow on the chair."*	{`GoToLocation`, `PutObject`}
	9.	*"Turn right and walk up to the couch. Pick up the red pillow off the couch. Turn around and walk ... to the chair."*	{`GoToLocation`, `PickUpObject`, `GoToLocation`}
	10.	*"Pick up the red pillow off the couch. Turn around and walk ... to the chair. Put the red pillow on the chair."*	{`PickUpObject`, `GoToLocation`, `PutObject`}
	11.	*"Turn right and walk up to the couch. Pick up the red pillow off the couch. Turn around and walk ... to the chair. Put the red pillow on the chair."*	{`GoToLocation`, `PickUpObject`, `GoToLocation`, `PutObject`}

Table 1: Example high level multi-intent (1.), low-level single-intent (2.–5.) and low-level multi-intent (6.–7.) tasks of type 'pick and place'. The model receives language commands (left col) together with relevant visual information, and predicts an intent (right col). Top/middle are from the original ALFRED dataset. Bottom instances from data augmentation.

	train	valid	test
# Commands	104,669	24,612	25,109
Percentage	70%	15%	15%

Table 2: Final data set statistics.

data set statistics are in Table 5 in the appendix.

4 Models

Our intent classification model took vector representations of the language command and visual context as input and predicted the underlying intent as one of 16 classes. We briefly describe the representation schemes for scene and language input. Afterwards, we present our proposed model, which extended a state-of-the-art language intent classifier to handle both visual and language input.

4.1 Visual Features

The visual data corresponding to a task instance in ALFRED dataset included the agent and object position information (Figure 2, Agent and Scene information). We represented the visual information of each task as a 4-dimensional vector with elements corresponding to (i) The L2 (Euclidean) distance between agent and target object, (ii) L2 distance between agent and receptacle object, (iii) L2 distance between target and receptacle object and (iv) the angle between the target object and the direction the agent is facing initially.

4.2 Language Features

We transformed the language command to pre-trained word embeddings (Pennington et al., 2014; Kenton and Toutanova, 2019; Peters et al., 2018). Specifically, we use Tok2Vec embeddings provided by SpaCy.[3] We mapped each word in an input command to its corresponding embedding and obtained a representation for the entire command by averaging the word embeddings. Following (Bunk et al., 2020) we augment the embeddings with word- and character-level n-grams.

4.3 The DIET Intent Classifier

DIET is a state of the art, natural language intent classification architecture developed for dialogue understanding tasks (Bunk et al., 2020). DIET classifiers are attractive for application to assistive technologies because they can be trained rapidly and work well even with small datasets. The DIET classifier represents natural language inputs as described above (Sec 4.2). This input representation is passed through a neural network transformer architecture (Vaswani et al., 2017) which is a state-of-the-art architecture for computing contextualized representations of input sequences. DIET is optimized to maximize the similarity between the final representation of the verbal command and an embedded representation of the true intent. We follow their optimization procedure, and at test time we predicted the intent with the closest predicted embedding to the gold label. We used the official implementation, with default parameters.[4]

4.4 Multi-modal DIET

We extended the DIET classifier to a multi-modal model (DIET-M) which predicted intents based on language and scene features. The language input was encoded exactly as in the original model. We then concatenated the output of the transformer along with the 4-dimensional numerical visual features and passed the result first through a 10% dropout layer, followed by two feed-forward layers of sizes 256 and 128 and finally through an output layer of size 40 to obtain a combined visual and language representation. ReLU was used as the activation function for all the feed-forward layers. This joint embedded representation was then used to identify the intents following DIET's original training objective, as described above.

5 Experiments

We present a series of experiments which assesses the impact of model complexity, multi-modal information as well as our data augmentation on final intent classification performance. This work focuses on robust multi-modal intent classification, and as such our experiments assume that the entity recognition and visual interpretation (such as object detection and location) have been solved externally. We discuss our contribution in the context of an end-to-end application Section 6.

PickUpObject}, {RotateAgent,PutObject}, {GoToLocation, PickUpObject, PutObject}, {RotateAgent, PickUpObject, PutObject}, {RotateAgent, PickUpObject, GoToLocation}, {RotateAgent, PickUpObject, GoToLocation, PutObject}

[3] https://spacy.io/usage/embeddings-transformers

[4] https://rasa.com/docs/rasa/reference/rasa/nlu/classifiers/diet_classifier/

5.1 Baselines

We compare DIET and DIET-M against a Multi-Layer Perceptron (MLP) with a single hidden layer. Two variations of the MLP were tested: (1) MLP which takes as input only the embedded language representations; and (2) MLP-M which is provided with the embedded language representations concatenated with the visual features, resulting in a multi-modal variant. Rectified linear unit (ReLU) was used as the activation function and stochastic gradient descent (Ruder, 2016) was used to minimize a cross-entropy loss. The output of the final layer was passed through a soft-max layer to get the probability distribution across all possible intents. At test time, the intent with the highest probability score was predicted as the true intent associated with a command.

We also report a simple majority class baseline, which labels all instances with the most prevalent class in the training set (`GoToLocation`).

5.2 Metrics

We report micro-averaged accuracy, acknowledging the class imbalance in our data set, as well as precision recall and F1 measure.

5.3 Results

Our experiments answered the following questions: (a) how important is the multi-modal (scene) input for accurate intent classification; (b) is a powerful contextual language encoding model necessary to achieve high intent classification performance; and (c) how does the training dataset augmentation impact performance with multi-intent commands? To answer the first question, we compared both machine learning models (DIET-M, MLP-M) against their unimodal language-only versions (DIET, MLP). To answer the second question, we compared the complex DIET classifier against the simpler MLP architecture, and a majority class baseline. Finally, the benefits of data augmentation were ascertained by testing DIET-M's performance on the same testing dataset after training on datasets with different levels of augmentation.

Powerful language encoders improve intent classification accuracy. Table 3 compares the performance of the majority class baseline (Majority), MLP and the DIET classifier. All models were trained and tested on the full, augmented data set. Unsurprisingly, we observed that all machine learning models outperformed the majority class baseline. Furthermore, the variants of the DIET classifier consistently achieved a higher score than the simpler MLP (improvement of 5.6% absolute accuracy). Even though both models achieve F1 measures > 90%, very high language understanding performance is essential for user satisfaction in dialogue systems in general, and in assistive technology settings in particular. In addition, our evaluation adopted "laboratory" conditions, assuming noise-free entity and vision processing. With these arguments in mind, and recalling the fact that DIET is by design fast and efficient, we conclude that state-of-the-art language understanding architectures are preferable for situated intent classification.

Method	Ac	Pr	Re	F1
Majority	0.142	0.142	1.0	0.248
MLP	0.451	0.379	0.374	0.333
DIET	0.591	0.409	0.508	0.429
MLP-M	0.929	0.931	0.929	0.930
DIET-M	**0.985**	**0.982**	**0.984**	**0.983**

Table 3: Intent classification performance of the majority class baseline, multi-layer perceptron (MLP) and our DIET classifier in a unimodal and multi-modal setup (-M). We report accuracy (Ac), precision (Pr), recall (Re) and F1-measure.

Grounding language in visual context information improved intent classification performance. Table 3 compares multi-modal model variants (DIET-M, MLP-M) – with access to visual *and* language information – against their unimodal variants, which classify intents based on language commands only and remain agnostic about the visual surroundings. For both the MLP and DIET we observed a substantial improvement with added visual information. This is unsurprising, given the fact that navigational language commands are often high-level and can only be fully disambiguated in the context of the environment. As evidenced by the large performance gain of our multi-modal models over their language-only counterparts, both systems successfully learned to leverage the additional visual context for accurate intent interpretation.

Data augmentation improved performance of DIET-M. We investigated the benefit of data augmentation on the best performing classifier (DIET-

Augmented	Ac	Pr	Re	F1
0%	0.630	0.529	0.607	0.545
10%	0.921	0.927	0.923	0.925
50%	0.952	0.947	0.944	0.945
100%	0.985	0.982	0.984	0.983
100% (multi)	0.981	0.974	0.976	0.973

Table 4: The performance of the DIET-M classifiers, trained on datasets with access to 0%, 10%, 50% or 100% of the augmented data. 100% (multi) tests only on the more challenging multi-intent subset of the test data.

M) by ablating the amount of augmented training data available to the classifier during training. Specifically, we augment 0%, 10%, 50% or 100% of the original ALFRED instances with multi-subtask variations (as described in Section 3) Rows 1-–4 in Table 4 show DIET-M performance trained on data sets with varying amounts of augmentation, and tested on the full, augmented test data. The model improved consistently with increased augmentation of the training data. Even a small amount of augmented data improved performance substantially, while more augmentation leads to diminishing returns. We finally analyzed specifically the benefit of data augmentation on understanding *multi-intent* commands, i.e., language commands which imply sequences of actions (bottom part of Table 1). To this end, we evaluated the classifier only on multi-intent commands. The result in the final row of Table 4 shows that the performance on these longer and more complex instances was practically on par with performance on the full test set, confirming that DIET-M successfully maps abstract comments to sequences of actions.

6 Discussion

We leveraged and extended a large-scale dataset of indoor navigation tasks to develop an intent classification component for robotic arm control to perform "pick and place" tasks. Our novel multimodal DIET classifier exceeded 98% in classification performance in an "in vitro" evaluation setup. We now discuss limitations of our work as well as future directions.

Toward end-to-end task completion. The intent classifier will be embedded in a larger system in order to enable end-to-end task completion. In our evaluation, we assumed that visual scene parsing (including object recognition and location) as well as entity recognition in the language had been solved perfectly and externally. In an ongoing project, the presented system is integrated with these components, leveraging the recent improvements and corresponding tools and frameworks powered by advances in machine learning, robotics and data sets (Liu et al., 2020; Zhu et al., 2020; Redmon and Farhadi, 2018). This paper presented a highly accurate system which provides a strong foundation and promising starting point for end-to-end integration as well as experiments under noisy conditions (e.g., malformed or ambiguous utterances, or speech recognition errors).

Diversity of tasks and inputs Our study was constrained to "pick-and-place" tasks which (a) are conceptually straightforward and (b) are typically expressed in a fairly regular, formulaic manner. Even though the underlying ALFRED data set was diverse and somewhat noisy due to its crowdsourced nature, future work will extend our scenario to more complex tasks. ALFRED includes a variety of tasks beyond "pick-and-place" and can directly support this line of work. Our way of constructing multi-intent subtasks by concatenating low-level descriptions biased the data towards long descriptions and an underrepresentations of co-referential pronouns (e.g., "Pick up the keys and put *them* in the bowl"). Future work could leverage a mix of human data collection and natural language generation from language models to further augment the training data.

The accuracy-flexibility trade-off. This work developed a highly accurate intent classifier motivated by the fact that efficient and reliable language understanding is paramount to effective human-robot interaction. To achieve this, we limited the scenarios to a single task type as well as a simple but inflexible intent classification task: We exhaustively enumerated possible intents as 16 classes, thus preventing the model from meaningfully classifying an input that does not correspond to one of these categories. A more flexible system would predict a *sequence* of atomic intent labels of varying length. To this end, the task could be reframed as multi-label classification; or a sequence-to-sequence model could be developed to translate a natural language input into a sequence of intent labels. Analyzing the trade-off between reliability

and flexibility in the context of robust multi-modal intent classification for assistive technologies is a fruitful direction for future research.

7 Conclusion

This paper presented a multi-modal intent classifier for "pick-and-place"-tasks which takes diverse natural language commands as input, and which will be incorporated into a natural language interface of an assistive robotic arm. Our work will help to improve the naturalness of human-robot communication, which to-date often consists of mechanical (joystick) control or formulaic and templated language input. We showed how a large-scale naturalistic data set for general indoor navigation can be adapted to support training of a specific, high-accuracy intent classifier. We extended a state-of-the-art natural language-based intent classifier to utilize both vision and language information. Our evaluation showed the effectiveness of our data augmentation, and the importance of *multi-modal* signal for our task. We hope that our work motivates a wider, cross-disciplinary use of large-scale naturalistic data sets – which are becoming more ubiquitous in the NLP and ML communities – as a valuable resource for developing flexible intelligent assistive technologies.

Acknowledgments

This research was supported by the ARC Industry Transformational Training Centre IC170100030 grant.

References

Brian S Armour, Elizabeth A Courtney-Long, Michael H Fox, Heidi Fredine, and Anthony Cahill. 2016. Prevalence and causes of paralysis—united states, 2013. *American journal of public health*, 106(10):1855–1857.

Philip Barry, John Dockery, David Littman, and Melanie Barry. 1994. Intelligent assistive technologies. *Presence: Teleoperators & Virtual Environments*, 3(3):208–215.

Yonatan Bisk, Deniz Yuret, and Daniel Marcu. 2016. Natural language communication with robots. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 751–761.

Tanja Bunk, Daksh Varshneya, Vladimir Vlasov, and Alan Nichol. 2020. Diet: Lightweight language understanding for dialogue systems. *arXiv preprint arXiv:2004.09936*.

David Chen and Raymond Mooney. 2011. Learning to interpret natural language navigation instructions from observations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 25.

Howard Chen, Alane Suhr, Dipendra Misra, Noah Snavely, and Yoav Artzi. 2019. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12538–12547.

Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 753–757, New Orleans, Louisiana. Association for Computational Linguistics.

Matthew Henderson, Iñigo Casanueva, Nikola Mrkšić, Pei-Hao Su, Tsung-Hsien Wen, and Ivan Vulić. 2020. ConveRT: Efficient and accurate conversational representations from transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2161–2174, Online. Association for Computational Linguistics.

Tan Kian Hou et al. 2020. Arduino based voice controlled wheelchair. In *Journal of Physics: Conference Series*, volume 1432, page 012064. IOP Publishing.

Brandi House, Jonathan Malkin, and Jeff Bilmes. 2009. The voicebot: a voice controlled robot arm. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 183–192.

Ripcy Anna John, Sneha Varghese, Sneha Thankam Shaji, and K Martin Sagayam. 2020. Assistive device for physically challenged persons using voice controlled intelligent robotic arm. In *2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, pages 806–810. IEEE.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. In *Interspeech 2016*, pages 685–689.

Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and Matti Pietikäinen. 2020. Deep learning for generic object detection: A survey. *International journal of computer vision*, 128(2):261–318.

Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. 2019. Benchmarking natural language understanding services for building conversational agents. In *10th International Workshop on Spoken Dialogue Systems Technology 2019*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Xavier Puig, Kevin Ra, Marko Boben, Jiaman Li, Tingwu Wang, Sanja Fidler, and Antonio Torralba. 2018. Virtualhome: Simulating household activities via programs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8494–8502.

Terrin Babu Pulikottil, Marco Caimmi, Maria Grazia D'Angelo, Emilia Biffi, Stefania Pellegrinelli, and Lorenzo Molinari Tosatti. 2018a. A voice control system for assistive robotic arms: preliminary usability tests on patients. In *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob)*, pages 167–172. IEEE.

Terrin Babu Pulikottil, Marco Caimmi, Maria Grazia D'Angelo, Emilia Biffi, Stefania Pellegrinelli, and Lorenzo Molinari Tosatti. 2018b. A voice control system for assistive robotic arms: preliminary usability tests on patients. In *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob)*, pages 167–172. IEEE.

Joseph Redmon and Ali Farhadi. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Sebastian Ruder. 2016. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.

Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. 2020. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10740–10749.

Simon Stepputtis, Joseph Campbell, Mariano Phielipp, Stefan Lee, Chitta Baral, and Heni Ben Amor. 2020. Language-conditioned imitation learning for robot manipulation tasks. *Advances in Neural Information Processing Systems*, 33.

Sumet Umchid, Pitchaya Limhaprasert, Sitthichai Chumsoongnern, Tanun Petthong, and Theera Leeudomwong. 2018. Voice controlled automatic wheelchair. In *2018 11th Biomedical Engineering International Conference (BMEiCON)*, pages 1–5. IEEE.

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*, pages 5998–6008.

Haidi Zhu, Haoran Wei, Baoqing Li, Xiaobing Yuan, and Nasser Kehtarnavaz. 2020. A review of video object detection: Datasets, metrics and methods. *Applied Sciences*, 10(21):7834.

A Dataset Statistics

	Intent	train	valid	test
{	GoToLocation }	14.3%	14.3%	14.3%
{	PickUpObject }	3.5%	3.7%	3.5%
{	PutObject }	3.5%	3.6%	3.5%
{	GoToLocation, PickUpObject }	7.2%	7.1%	7.2%
{	PickUpObject, GoToLocation }	3.5%	3.6%	3.6%
{	GoToLocation, PutObject }	7.1%	7.1%	7.1%
{	PickUpObject, PutObject }	3.5%	3.6%	3.6%
{	RotateAgent, PickUpObject }	3.6%	3.4%	3.7%
{	RotateAgent, PutObject }	3.6%	3.5%	3.6%
{	GoToLocation, PickUpObject, GoToLocation }	7.1%	7.1%	7.1%
{	PickUpObject, GoToLocation, PutObject }	7.0%	6.9%	7.2%
{	GoToLocation, PickUpObject, PutObject }	7.3%	7.1%	7.2%
{	RotateAgent, PickUpObject, PutObject }	3.6%	3.5%	3.5%
{	RotateAgent, PickUpObject, GoToLocation }	3.6%	3.6%	3.6%
{	GoToLocation, PickUpObject, GoToLocation, PutObject }	14.2%	14.3%	14.2%
{	RotateAgent, PickUpObject, GoToLocation, PutObject }	7.2%	7.4%	7.0%
	Total Commands	104,669	24,612	25,109
	Total Percentage	70%	15%	15%

Table 5: Full distribution of task instances by intent type in our final data set.