PropBank-Powered Data Creation: Utilizing Sense-Role Labelling to Generate Disaster Scenario Data

Mollie Shichman[1], **Claire Bonial**[2], **Taylor Hudson**[3],
Austin Blodgett[2], **Francis Ferraro**[4], **Rachel Rudinger**[1]
[1]University of Maryland College Park, [2]Army Research Lab,
[3]Oak Ridge Applied Universities, [4] University of Maryland Baltimore County
{mshich, rudinger}@umd.edu, claire.n.bonial.civ@army.mil

Abstract

For human-robot dialogue in a search-and-rescue scenario, a strong knowledge of the conditions and objects a robot will face is essential for effective interpretation of natural language instructions. In order to utilize the power of large language models without overwhelming the limited storage capacity of a robot, we propose PropBank-Powered Data Creation. PropBank-Powered Data Creation is an expert-in-the-loop data generation pipeline which creates training data for disaster-specific language models. We leverage semantic role labeling and Rich Event Ontology resources to efficiently develop seed sentences for fine-tuning a smaller, targeted model that could operate onboard a robot for disaster relief. We developed 32 sentence templates, which we used to make 2 seed datasets of 175 instructions for earthquake search and rescue and train derailment response. We further leverage our seed datasets as evaluation data to test our baseline fine-tuned models.

Keywords: PropBank, Object Affordances, Synthetic Data Creation, Fine-tuning

1. Introduction

In dangerous and dynamic problem spaces like search and rescue, instructing a robot agent in the field via natural language offers a flexible means of communication with a low cognitive burden on rescue workers. However, it is imperative that the robot agent be able to correctly understand and execute natural language instructions from its human operator. For example, for the instruction “move past the chair and try to find an entrance,” the robot agent should be able to determine if the instruction is related to navigation, interacting with objects with a mechanical arm, identifying obstacles in its environment, or a combination of those options. These instructions are often specific to the disaster scenario in question, the tools required for search and rescue for the given disaster, and the overall environment where the disaster occurred. Finally, the robot agent needs physical common-sense reasoning to effectively follow instructions in such a precarious environment.

Large language models (LLMs) have shown great promise for encoding world knowledge (Petroni et al., 2019), as well as strong performance on instruction following tasks (Ouyang et al., 2022; Wang et al., 2022; Chung et al., 2022). However, these models have drawbacks for human-robot interaction in disaster relief. Instruction LLMs are often unspecialized, aimed at accomplishing a plethora of diverse written tasks rather than specializing in a domain-specific task with its own assumptions and peculiarities. Additionally, LLMs are trained on tasks that do not require a strong basis in physical common sense, including the potential usages of objects, which we term ’affordances.’ As a LLM may not have any specific semantic training, it is unclear how they will perform on relevant semantic scenarios like reasoning about properties of objects. Another challenge is that LLM’s reasoning can be difficult to interpret and predict.

Furthermore, there is a pragmatic limitation of available hardware in robot systems. As LLMs vastly increase in size, it becomes more difficult for smaller hardware systems to use these models. Most robots use one commercially available GPU, and assuming the GPU has 24 GB of memory and the LLM is using 4-bit quantization (Dettmers et al., 2023), the robot could realistically only run an LLM with 40B parameters. A robot working in disaster relief needs many other systems onboard, so memory space is even further limited down to smaller 7 billion or 13 billion parameter models. These smaller models would need fine-tuning to be competent in the field due to their size. However, fine-tuning data for specific types of disasters are not easily available.

We hypothesize a solution to this problem space is to fine-tune small LLMs with a wide variety of disaster-specific data. These LLMs should be able to answer both multiple choice and open ended questions about how to execute different subtasks of the disaster. They should be able to reason about the various objects a robot could come across during a disaster relief mission. This includes knowing the functions of different objects, the different states an object can be in, the relative size and shape of objects, etc. Yet another important task is recogniz-

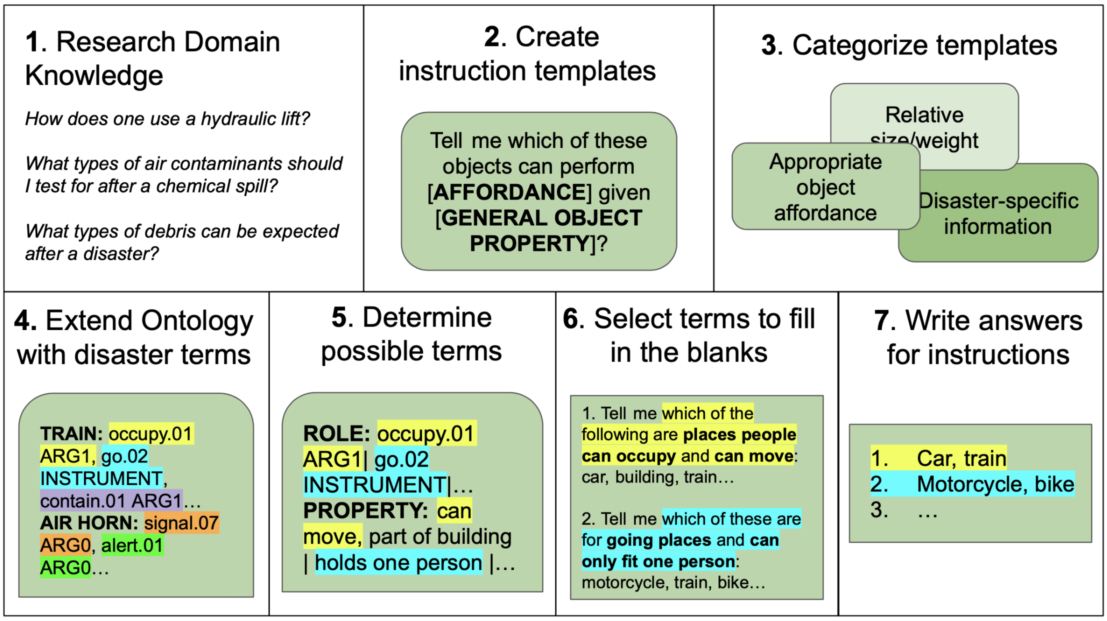

1. Research Domain Knowledge
How does one use a hydraulic lift?
What types of air contaminants should I test for after a chemical spill?
What types of debris can be expected after a disaster?
2. Create instruction templates
Tell me which of these objects can perform [AFFORDANCE] given [GENERAL OBJECT PROPERTY]?
3. Categorize templates
Relative size/weight
Appropriate object affordance
Disaster-specific information
4. Extend Ontology with disaster terms
TRAIN: occupy.01 ARG1, go.02 INSTRUMENT, contain.01 ARG1…
AIR HORN: signal.07 ARG0, alert.01 ARG0…
5. Determine possible terms
ROLE: occupy.01 ARG1| go.02 INSTRUMENT|…
PROPERTY: can move, part of building | holds one person |…
6. Select terms to fill in the blanks
1. Tell me which of the following are places people can occupy and can move: car, building, train…
2. Tell me which of these are for going places and can only fit one person: motorcycle, train, bike…
7. Write answers for instructions
1. Car, train
2. Motorcycle, bike
3. …

Figure 1: The workflow for generating gold-standard instructions. After collecting domain knowledge about different types of questions to be answered, we created templates for the different types of instructions and categorized them to ensure a relatively even distribution of queried knowledge in our results. We then determine the terms, roles, and/or vocabulary that could fill in the templates. Creating these templates allowed us to quickly generate gold standard instructions for object affordances and earthquake search and rescue. These instructions were then used for both perturbing the embeddings of a language model during the training data generation stage and evaluating the resulting fine-tuned model. Instructions corresponding to the occupy.01 ARG1 role are highlighted in yellow. Instructions corresponding to the go.02 INSTRUMENT role are highlighted in blue. More in-depth examples of seed sentences can be found in table 1.

ing what objects have the potential to be dangerous. All of these functionalities are necessary in order for successful human-robot interaction in these disaster scenarios, both for ease of interaction and for the robot agent's successful execution of the instruction. The goal of this work is to create a framework for generating data that can provide a basis for reasoning about this wide variety of tasks. This process can be seen in Figure 1.

While the tasks we want an LLM to accomplish are diverse and ambitious, Taori et al. (2023) has had great success with a similar task to ours. They instruction fine-tuned the LLaMa 7B model to have similar instruction following performance to GPT3.5, a much larger LLM. To do this, they expertly crafted seed instructions that were fed into OpenAI's `text-davinci-003` as In-Context Learning (ICL) for generating high-quality synthetic data (Dong et al., 2023). While effective, their methodology for creating seed sentences for synthetic data generation is not appropriate for our use case for two reasons. For one thing, the seed instructions used by Taori et al. (2023) were created by a group of experts whose broad domain and lack of time constraints meant they could generate uniquely formatted seed instructions on a relatively ad hoc basis. We need a systematized pipeline to ensure that our sentences are generated quickly as well as accurately, and that all relevant areas of our disaster domain are covered by our seed sentences. This is so a robot agent can be deployed quickly and with high accuracy for disasters that place time constraints on when relief efforts must happen. Additionally, the seed instructions were not based in any particular semantics that Taori et al. (2023) wanted their model to "understand", while we need our model to have semantic understanding of the disaster and the objects a robot agent could encounter while navigating it.

To solve these issues, we propose an expert-in-the-loop data generation pipeline called PropBank-Powered Data Creation, which can be seen in Figure 1. In this pipeline, seed sentences are informed by disaster expert knowledge, then created by a linguistic expert in one work day. These seed sentences are then used as in-context learning for synthetic data generation to produce a much larger dataset than would otherwise be possible with a tight timeframe and a highly specialized domain. The seed sentences are constructed using templates rooted in the semantic properties of disaster-relevant senses from the PropBank lexicon (Palmer et al., 2005). These seed instructions also serve as a semantically informed evaluation, since they are not included in the resulting synthetic dataset.

The contributions of this paper are as follows:

1. A process where linguists, with minimal disaster expert input, can quickly generate gold-standard seed sentences to be used during synthetic data generation. This includes 35 sentence templates for generating seed sentences.

2. An ontology of over 300 disaster relevant vocabulary terms that are annotated with PropBank sense-role labels representing the objects' affordances and change of state potentials

3. Two sets of 175 seed sentences: one focused on earthquakes, and one focused on the Ohio Train Derailment[1]

2. Background

In the sections to follow, we provide background information on the source of common-sense object affordance knowledge that we leverage to seed the generation of fine-tuning data, followed by the fine-tuning procedure we adopt.

2.1. Object Properties

As interaction with objects is a major component of the instructions a robot may be given, it is important to have a framework for describing different types of objects and what affordances, or functionalities, a given object may have, as well as the canonical changes of state the object may undergo.

We leverage the Affordance Ontology of disaster-relevant vocabulary terms (Shichman et al., 2023) that adopts a PropBank-style (Palmer et al., 2005) representation of the vocabulary's function and state changes in terms of semantic roles each term played with respect to an event. This resource, an extension of the Rich Event Ontology (Bonial et al., 2021), is a hub mapping event concepts from different semantic role labeling resources and includes "qualia relations," and specifically "telic" relations that denote the affordances of objects in terms of events (Kazeminejad et al., 2018).

The Rich Event Ontology previously only represented a limited number of telic qualia relations expressed between objects and particular events. The Affordance Ontology extends the vocabulary and representations of the Rich Event Ontology by representing object affordances in terms of PropBank sense-role pairings for given senses of events. For example, within the Affordance Ontology, the affordance of a bucket is labeled as an ARG0, or "container" of a *contain.01* event, defined loosely as "hold inside."[2] A box would not only be represented with this same containing affordance, but would also be characterized by a representation of a canonical change of state: to be open (ARG1 of *open.01*) or closed (ARG1 of *close.01*).

The Affordance Ontology provides a basis of a vocabulary of objects that are likely to be present in generic search and rescue scenarios. This means that this resource can serve as a gold-standard set of object properties within our disaster use cases. In this research, we not only use the Affordance Ontology, but also extend it to new objects and affordances leveraging our PropBank-Powered Data Creation workflow (described in detail in section 3).

There are other resources for defining object functionality that we considered for our application—notably the Suggested Upper Merged Ontology (SUMO) (Niles and Pease, 2003), which includes axioms and object definitions to indicate object affordances. However, we preferred to use PropBank because of its elegance in representing the object's functionality and because of the amount of data supporting its approach. Furthermore, SUMO is more focused on connecting semantic concepts stored on the word level rather than fully describing events. Using PropBank, specifically the PropBank rolesets, also allows for our work to be integrated with other Natural Language Understanding resources like Abstract Meaning Representation, which shares the same roleset representation of events (Banarescu et al., 2013) and can distill instructions into action primitives and their corresponding parameters (Bonial et al., 2020).

2.2. Generating Natural Language Instructions

Obtaining high quality language for training and fine-tuning language models is expensive and time consuming. With the rise and improvement of LLMs, significant work is being done to examine if LLMs can do this work with more speed and with the same level of accuracy as crowd-sourcing.

Notably, Wang et al. (2022) developed a framework for prompting a language model to create a diverse set of instructions which could be used to fine-tune said language model. Specifically, the process begins with writing 175 unique seed instructions, then prompting GPT3 to generate a new set of diverse instructions, then filtering out instructions of insufficient quality via ROUGE-L score. Af-

[1] https://www.reuters.com/world/us/ohio-carry-out-controlled-release-chemicals-train-derailment-site-2023-02-06/

[2] https://propbank.github.io/v3.4.0/frames/contain.html

ter generating approximately 52,000 instructions, these instructions were then fed back into GPT3 for fine-tuning. This resulted in SELF-INSTRUCT, a fine-tuned GPT3 model that humans rated significantly better on instruction tasks than vanilla GPT3. Furthermore, though it performed worse than all versions of InstructGPT, it was close and still competitive, and required much less human labor (Wang et al., 2022).

Inspired by the success of Wang et al. (2022) and the release of LLaMa (Touvron et al., 2023), Taori et al. (2023) created their own fine-tuned instruction-following model, Alpaca. Alpaca largely followed the same algorithm for generating their own instructions as SELF-INSTRUCT. The major innovation of Alpaca was that it used the output of GPT3 to fine-tune the smaller LLaMa 7B model rather than GPT3 itself. This provided a major performance boost, with humans rating the Alpaca answer to be the preferred one just as often as Vanilla GPT3. We follow the approach of Alpaca, but make use of PropBank to quickly develop seed instructions.

3. PropBank-Powered Data Creation Methodology

To quickly turn expert knowledge from both written and oral sources into a disaster-specific LLM, we aim to develop an efficient way of generating a set of gold standard seed instructions. These seed sentences will then be used as in-context learning for synthetic data generation, which in turn will be used to fine-tune a smaller LLM to enhance its performance on a specific disaster domain.

To create the initial set of seed sentences, we developed the PropBank-Powered Data Creation Pipeline, which relies upon sentence templates with slots that are populated largely by object vocabulary from the Affordance Ontology (Shichman et al., 2023). The vocabulary that can be used within a particular slot is constrained by the PropBank-style representation of properties such as its affordances and change of state potentials. For example, to create a seed sentence querying relative weight, one would take the template “Which of these objects is the lightest? [LIST OF OBJECTS]” and fill in the “blank” with a list of objects that were randomly generated, then refined to only include objects with differentiable weights. Template examples can be seen in Table 1. More complex and elaborate examples can be found in Table 2.

Thus, templates can be semi-automatically populated based on linguistic properties of the template slot, instead of having disaster experts develop dozens of unique instructions. This decreases time to robot deployment while maintaining the accuracy of the seed sentences. The challenge therefore becomes how to effectively template important properties for downstream use.

3.1. Creating the Templates

To tackle the challenge of creating templates for generating seed sentences, we developed an annotation workflow in which graduate student linguistic annotators brainstormed a variety of instructions and questions that a disaster-relief specialist might want a robot to be able to execute or answer. The annotators were instructed not to write instructions outside of a LLM’s capabilities, like image identification or referring to a 3D space the LLM cannot perceive (e.g. "Get that can from your right"). Some examples of brainstormed questions include “What can be used for travel and carry large loads?” and “How can an adult reach the ceiling?”.

The linguistic annotators then moved from the hypothetical to real data by incorporating disaster expert knowledge. For the purposes of this paper, our ‘expert knowledge’ came from written documents about the response to the Ohio train derailment (Air Sampling; Water Sampling; Soil Sampling; Derailment Tools; Yan et al., 2023) and the search and rescue process after earthquakes (Arranz et al., 2023; Hydraulic Rescue Tools; Scarbury, 2015; Thermal Cameras). A separate author collected the expert knowledge, and our annotators reviewed these data before constructing the query templates. Our queries were focused on a few key pieces of disaster information. we gathered expert data about the specific subtasks each disaster had. For example, for earthquakes, we researched how to lift and remove rubble from a building collapse, and for the train derailment the annotators queried about the types of environmental testing that were done to detect dangerous chemicals in the area. We also queried about the specific objects used in each subtask, what they are used to achieve, and how to use them safely. Third, we researched precautions that should be taken for the disaster as a whole, both by civilians and by rescue workers. Without this expert knowledge, the templates would not be as useful or cover all relevant information. Examples of the resulting disaster-related questions that came from this research are in step 1 of Figure 1.

The annotators then inspected all of the brainstormed instructions, generalized over them, then wrote original instruction templates, as exemplified in step 2 of Figure 1. For the example “What can be used for travel and carry large loads,” the central notion (here, of having a task (travelling) that needs completion with the help of an object (a type of vehicle with the affordance of *go.02* INSTRUMENT) that has additional constraints that go beyond the basic affordance label (ability to carry large loads) was then “templatized” into prompts of the form, *Tell me which of these can perform [AFFORDANCE] given*

Category	Templates	Examples	Instances in Seed Sets
Relative Size/weight	Biggest Object, Heaviest Object, Relative Fit	Which of these objects is the lightest? outlet, broom, pail, orange, screen Would a shoe fit in a bag?	15
Appropriate Object Affordance	Basic Affordance, Size Restricted, Shape Restricted, General Property Restricted, Goal Restricted, Difference within Affordance, Difference within Affordance given Criteria	Which of the following can be used to climb and is bigger than a table? stile, stairway, stepladder, step, ladder What should I use if I want to learn something from the internet? What is the difference between a window and a pane?	38
Is-A and Hypernyms	Basic Is-A, Identical Usage, Sub-Types	Can you use a shed as a barn? List several types of truck and their use cases.	16
Objects in Risky Situations	Cause Injury, Cause Danger, Cause Object Damage	Which of the following objects would be the most dangerous if it hit something? dvd, screen, wall, drum, mat	16
Required Equipment	How to Use, Equipment for Scenarios, Role of Equipment in Task	Give a step by step explanation of how to use a concrete saw. What role does an air canister play in testing air quality?	15
Primary and Secondary Object Facts	Where Object Found, Objects in Location, Secondary Uses, Frequency of use, Average Knowledge of Use, Ease of Interaction Given Object State	Hey, which of the following can be used as a lever? art, motorcycle, picture, dvd, broom How well does the average person know how to use a concrete saw? Is a raised or lowered drawbridge more effective at getting cars across the river?	34
Disaster Specific Knowledge	Preparations, Warning Signs, General Information	List and explain the different hazards to look out for besides train cars after a train derailment.	10
Instruction Following	Instruction Identification, Follow-Up Questions	Choose the navigation instruction: drink from the bottle, sail a boat, enter the doorway	30

Table 1: An overview of the types of templates within each category, some examples of resulting seed sentences within each category, and the number of instances of each category within the resulting seed dataset. Note the emphasis on affordances, object knowledge, and instruction knowledge.

[GENERAL OBJECT PROPERTY]?. We then categorized this resulting template under the general category of "Appropriate object affordances" alongside other template instructions focused on querying about objects' functionalities and affordances (see step 3 of Figure 1). The complete list of template categories with corresponding examples can be found in Table 1.

After developing the templates, the annotators used a list of objects from the disaster-specific expertise and labelled each object with all applicable PropBank sense-role pairings. We added these labels to Affordance Ontology previously described in 2.1. For instance, "Train," which is relevant to the Ohio train derailment, was labelled *occupy.01* ARG1, *go.02* ARG2, and *contain.01* ARG0 by our annotators. This means a train can hold people, be used for transporting people, and can contain objects. "Air horn," which is relevant to Earthquake search and rescue, was labelled with *signal.02* ARG0 and *alert.01* ARG1, meaning that an air horn can both signal information and warn of potential danger. This extension of the Affordance Ontology can be seen in step 4 of Figure 1. Examples of how Affordance Ontology labels connect to vocabulary used in the templates are in Table 2.

3.2. From Templates to Seed Instructions

For our next step, we determined what vocabulary could potentially fill in the blanks for each template. We examined each template and determined which vocabulary terms with associated linguistic properties from the list could appropriately fill in the blanks of each instruction. For instance, we determined that the affordance of *occupy.01* ARG1 (i.e. an object that a human can occupy) can appropriately fill in the AFFORDANCE slot for the template *Tell me which of these objects can perform [AFFORDANCE] given [GENERAL OBJECT PROPERTY]*. We then chose properties corresponding to each chosen sense-role label to fill in the GENERAL OBJECT PROPERTY slot, thus further restricting the number of correct answer objects. This process is shown in step 5 of Figure 1, where one exemplified PROPERTY slot associated with *occupy.01* ARG1 is *can move*, which restricts the list of potential correct answers from *balcony, barn, boat, building, car, floor (story), house, truck, train* to be *boat, car, truck, train*. Another exemplified property slot associated with *go.02* INSTRUMENT is *holds one person*, which restricts the resulting correct answers with the *go.02* INSTRUMENT affordance to only *motorcycle, bike*. This process of choosing appropriate affordances and properties for the Identical Use Case template is shown in Table 2.

We chose all possible vocabulary terms with associated linguistic properties for each template, then randomly selected which vocabulary items would fill in a particular blank to generate the final seed questions. An example of a final seed instruction, arising from the template "Tell me which of these objects can perform [AFFORDANCE] given [GENERAL USE CASE]" is "Tell me which of the following are places people can occupy and can move: car, building, train.". The resulting gold-standard instruction is seen in step 6 of Figure 1.

The linguistic annotators each decided on the correct answers based on context. For disaster related knowledge and required equipment knowledge, the annotators relied heavily on our disaster expert sources. In general, answers could not be automatically generated from templates because we often tested for linguistic knowledge that went more in-depth than the knowledge encoded in PropBank sense-role affordance labels. One example is in step 7 of Figure 1. Objects that have the label *occupy.01* ARG1 cannot be differentiated by mobility by affordance label alone. Similarly, in Table 2, sharing an affordance of *store.01* ARG2 does not indicate or preclude that "barn" and "shed" have a hypernym or is-a relationship. The annotators had to use their own common-sense capabilities to achieve the level of granularity we need for assessing LLM common sense capabilities.

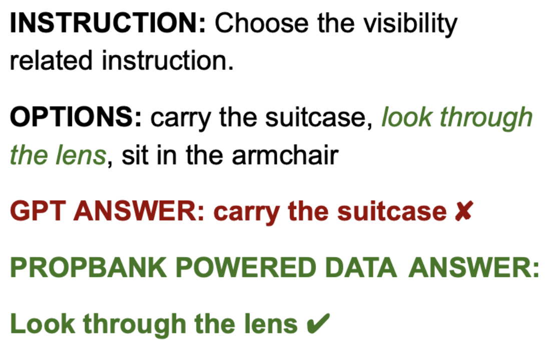
Upon request, we will make available both complete sets of seed questions, which also serve as an evaluation set for the model tuned for an earthquake disaster and the Ohio train derailment, respectively. In Table 2, we demonstrate our workflow for developing the disaster-specific seed set efficiently for the Identical Use Case template. With our annotation workflow for developing new models for new disaster scenarios, we can use an expert's time to provide only disaster-specific questions and vocabulary, as well as rating existing template quality.

		Populated by...
Template	Can you use [object-slot1] as a/n [object-slot2]?	Two objects w/ identical affordance
Potential Slot 1 Affordances	Path-of enter.01	**doorway**, **opening**, **gateway**, entrance, etc.
	ARG2 of store.01	**shed**, **barn**, **greenhouse**, silo, etc.
	Path-of go.02	**road**, **train track**, **floor**, doorway, trail, etc.
Potential Slot 2 Affordances	*same as above*	*same as above*
Seed 1	Can you use a **doorway** as an **opening?**	
Answer 1	**Yes** because a doorway is a type of opening found in buildings.	
Seed 2	Can you use a **shed** as a **barn?**	
Answer 2	**No** because a shed is too small to store hay, livestock, and tractors like a barn can.	

Table 2: Population of templates leveraging semantic role labeling linguistic features for quick generation of domain-specific seed sentences: The template requires two objects within affordances that annotators identified contain terms with hypernym relationships. Two objects with the same sense-role label, or affordance, are then randomly selected to fill each slot, and a linguistic annotator uses common sense knowledge to answer the resulting query. By training the model on both correct and incorrect answers that naturally arise from random generation, the deeper linguistic meaning of use-case hypernyms is expressed in our data.

4. Resulting Datasets

Our resulting datasets balance between covering a wide variety of physical object properties, such as size and weight, and holding specific knowl-



INSTRUCTION: Choose the visibility related instruction.

OPTIONS: carry the suitcase, *look through the lens*, sit in the armchair

GPT ANSWER: carry the suitcase ✘

PROPBANK POWERED DATA ANSWER:

Look through the lens ✔

Figure 2: An example of output from our preliminary model developed using the earthquake PropBank Powered Data Creation dataset. Here, `text-davinci-003` (A version of GPT 3.5) fails to choose the correct instruction from the options, but our much smaller model with PropBank Powered Data Creation can successfully correlate visibility with the pertinent instruction.

edge for an LLM to draw from when generating synthetic data based on the dataset. Furthermore, the datasets thoroughly cover required information for two very different types of disasters. For earthquakes, the priority is rescuing trapped individuals and clearing away rubble and partially collapsed buildings. For the Ohio train derailment, the focus was on monitoring the air, water, and soil for dangerous chemicals and ensuring the volatile chemicals that leaked from the train cars did not explode.

We initially tested PropBank-Powered Data Creation with our earthquake seed sentence dataset. This was a lengthy process of determining the types of templates we wanted, what they would be, and what vocabulary fit with each template. In contrast, developing the seed sentences for the Ohio train derailment took about 10 hours because we built on the pre-existing templates and potential choices for each fill in the blank. We are now confident that a disaster expert would need to give an hour of their time and some pointers to relevant literature to make PropBank Powered Data Creation successful. An expert annotator would then need one work day to develop the seed sentences. This means that the time between interviewing the disaster expert and deploying a model using PropBank-powered data could be as little as 3-4 days, depending on computational fine-tuning resources.

5. Baseline Fine-Tuned Model

The next step in our research is to use the PropBank-powered data as in-context learning examples for generating a synthetic dataset that will, in turn, fine-tune a small language model. We have made a preliminary model using the PropBank-powered earthquake data as our seed sentences, `text-davinci-003` as the model that generated a synthetic dataset of 20,000 instructions (OpenAI, 2023), and the LLaMa 7B model for fine-tuning (Touvron et al., 2023). We then had evaluators with expertise in linguistics compare the outputs of `text-davinci-003` and our PropBank-powered model by voting for which LLM won or if there was a tie and rating the quality of the winning answer on a scale of 0-3.

While the model our team developed had some successes, as can be seen in Figure 2, our preliminary results show we still have work to do. We had 3 annotators vote in our head-to-head testing, which resulted in our model winning approximately 8% of the evaluation prompts, tying with `text-davinci-003` for approximately 22.5% of the prompts, and losing to `text-davinci-003` for approximately 69% of the prompts. Further investigation found this was likely due to poor alignment between the seed sentences and the synthetic data. We believe the poor alignment was due to insufficient in-context learning during the data generation process, and are looking to improve this in future iterations. Making a preliminary model did prove that PropBank Powered Data Creation can be used both as evaluation and as seeding data, and we are excited to explore those capabilities as well in future work.

6. Related Work

6.1. Evaluation Datasets for Robots

Ahn et al. (2022) tests LLMs' abilities to execute instructions by developing a set of tasks for the robot agent to learn using reinforcement learning, then training a model to calculate the probability of a task being completed successfully paired with the probability that a natural language instruction will precede a given task. To do this, the authors wrote 101 instructions addressing various degrees of semantic complexity, including following primitive instructions, abstract nouns and verbs, and long-horizon planning that requires many steps to accomplish the instruction. The model, called SayCan, developed skills that transfered from the mock kitchen where it was trained to a real kitchen with minor losses in planning and performance. More interestingly, the authors also showed that SayCan performed better when they used larger LLMs with more linguistic knowledge. They also were able to utilize chain-of-thought fine-tuning to get a natural language explanation about the tasks that SayCan executed in order to fulfill the instruction.

Rather than having the LLM create a policy for a robot agent to execute itself, Xie et al. (2023) have GPT 3.5 translate the premise of the instruction from natural language to Planning Domain Definition Language (PDDL), an explicit way of defining all objects, predicates, and available actions within an environment. To test GPT 3.5's abilities to translate tasks, the authors developed tasks related to block stacking and navigating a kitchen that test an LLM's basic parsing competence, object association between natural language and entities in PDDL, numerical reasoning, physical and spatial reasoning, and world knowledge. They found that GPT 3.5 was able to perform well when the instructions were completely explicit and had decent performance at filling in the blanks for specifying goals and had decent reasoning about basic real world objects and relations. However, the authors also found that GPT 3.5 could not handle the complex and ambiguous physical relationships, and that GPT 3.5 likely relied extensively on the one-shot example it as given, rather than reasoning about the domain as a whole (Xie et al., 2023).

6.2. Robots and Language Models

PaLM-E is a multi-modal model designed to accept image, text, and sensor data and then output images, answers, or plaintext robot policy (Driess et al., 2023). This is achieved by vectorizing images into the same space as text embeddings, which allows for multi-modal fine-tuning but makes it unclear how the model would determine a particular robot's capabilities. RT-2 takes PaLM-E a step further by encoding language, vision, and actions into the same embedding space (Brohan et al., 2023). This allows for the robot agent to go beyond making only policy to making specific moves.

Instruct2Act takes a different approach and trains a LLM to output python code for a closed loop of perception, planning and actions (Huang et al., 2023). It does this by supplying the LLM with a variety of APIs for completing perception and action tasks. The scope of testing was limited to table top simulations, but the framework is inherently more flexible because the model can be fine-tuned to produce different python code.

These models all elicit interactions with the physical world, but Ghaffari and Krishnaswamy (2023) argue that these connections can't fully capture the complexity of the physical world because they don't include any physical data beyond images. To solve this problem, they train a neural network on physical simulations, then create a LLM embedding affine transformation matrix from both the physical embedding space and GPT3 embeddings. They find that LLM embeddings in the physical embedding space do correlate with the objects they describe, Most interestingly, nouns have a stronger correlation, and are thus more grounded, than verbs and attributes, much like how nouns are often learned first during language acquisition (Ghaffari and Krishnaswamy, 2023).

7. Future Work

In addition to our immediate goal of improving synthetic data generation techniques and fine-tuning parameters, we are interested in expanding PropBank Powered Data Creation to become multimodal. While even smaller multi-modal models are still too large to be useful in our robotics domain, there is a clear path for the expansion of our protocol. Notably, we hope to gather image data that can reinforce what different objects may look like in a given environment, how to interact with relevant equipment, and objects performing their affordances or changing states. These images could be paired with PropBank labels, vocabulary terms, and complete instructions. The variety of ways images can be combined with PropBank Powered Data Creation makes this an exciting new avenue for improving transformer model performance on disaster scenarios.

8. Conclusion

We introduce PropBank Powered Data Creation, a pipeline for efficiently creating semantically motivated seed sentences to be used for generating synthetic data for disaster related scenarios. We extended our Affordance Ontology and created 2 sets of 175 seed sentences for the domains of earthquake search and rescue and chemical spills following train derailments. These seed sentences extensively query objects' affordances, physical characteristics, changes of state, and fine-grained properties to ensure thorough evaluation of a LLM trained on PropBank Powered Data Creation-based synthetic data. We created a LLM demonstrating this full pipeline, and will continue to work on aligning our synthetic data to our seed sentences to increase LLM performance in disaster-related domains.

9. Ethical Considerations

PropBank Powered Data Creation is fundamentally based on biasing a language model towards feedback from a small group of selected sources. While this is for a positive effect within our domain, it may be harmful in domains that require more social common sense than ours. Within our templates, we tried as much as possible to be gender-neutral to discourage gender bias.

Our biggest form of bias is in assumptions of the specifics of our objects. We imagined our objects from a Western perspective, which can affect the affordances assigned to the object and how we query the object's properties. For instance, we imagine "curtains" to be window dressings, but in nomadic cultures a curtain could be used to separate living spaces within a tent. A positive about the structure of PropBank Powered Data Creation is that it purposefully allows time for adding and editing to the Affordance Ontology in order to align the data to a particular disaster and location. However, this is time consuming and puts the onus on the linguistic annotator to adjust the ontology both quickly and with cultural sensitivity.

Though the domain of this project is robots in disaster relief scenarios, we have not tested any implementation of this dataset on a robot, let alone a robot in a dangerous situation. We caution that extensive grounded testing must be done on any LLM resulting from these data before any real-world implementation can occur safely.

10. Works Cited

Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Daniel Ho, Jasmine Hsu, Julian Ibarz, Brian Ichter, Alex Irpan, Eric Jang, Rosario Jauregui Ruano, Kyle Jeffrey, Sally Jesmonth, Nikhil J Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Kuang-Huei Lee, Sergey Levine, Yao Lu, Linda Luu, Carolina Parada, Peter Pastor, Jornell Quiambao, Kanishka Rao, Jarek Rettinghouse, Diego Reyes, Pierre Sermanet, Nicolas Sievers, Clayton Tan, Alexander Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Mengyuan Yan, and Andy Zeng. 2022. Do as i can, not as i say: Grounding language in robotic affordances.

Air Sampling. 2024. Air sampling data.

Adolfo Arranz, Simon Scarr, and Jitesh Chowdhury. 2023. Searching for life in the rubble: How search and rescue teams comb debris for survivors after devastating earthquakes.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, pages 178–186.

Claire Bonial, Susan W Brown, Martha Palmer, and Ghazaleh Kazeminejad. 2021. The rich event ontology. *Computational Analysis of Storylines: Making Sense of Events*, page 47.

Claire Bonial, Lucia Donatelli, Mitchell Abrams, Stephanie Lukin, Stephen Tratz, Matthew Marge, Ron Artstein, David Traum, and Clare Voss. 2020. Dialogue-amr: abstract meaning representation for dialogue. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 684–695.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil J. Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael S. Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong T. Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. 2023. RT-2: vision-language-action models transfer web knowledge to robotic control. *CoRR*, abs/2307.15818.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Derailment Tools. 2022. Derailment response.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. 2023. A survey on in-context learning.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. 2023. Palm-e: An embodied multimodal language model. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 8469–8488. PMLR.

Sadaf Ghaffari and Nikhil Krishnaswamy. 2023. Grounding and distinguishing conceptual vocabulary through similarity learning in embodied simulations. In *The 15th International Conference on Computational Semantics*.

Siyuan Huang, Zhengkai Jiang, Hao Dong, Yu Qiao, Peng Gao, and Hongsheng Li. 2023. Instruct2act: Mapping multi-modality instructions to robotic actions with large language model.

Hydraulic Rescue Tools. 2022. Hydraulic rescue tools: What are your options?

Ghazaleh Kazeminejad, Claire Bonial, Susan Windisch Brown, and Martha Palmer. 2018. Automatically extracting qualia relations for the rich event ontology. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2644–2652.

Ian Niles and Adam Pease. 2003. Linking lexicons and ontologies: Mapping wordnet to the suggested upper merged ontology. In *Ike*, pages 412–416.

OpenAI. 2023. Openai gpt-3 api [text-davinci-003].

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Matt Scarbury. 2015. Rescue methods structural collapse step cutting concrete.

Mollie Shichman, Claire Bonial, Austin Blodgett, Taylor Hudson, Francis Ferraro, and Rachel Rudinger. 2023. Use defines possibilities: Reasoning about object function to interpret and execute robot instructions. In *Proceedings of the 15th International Conference on Computational Semantics*, pages 284–292, Nancy, France. Association for Computational Linguistics.

Soil Sampling. 2024. Soil and sediment sampling data.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca. Technical report, Stanford University.

Thermal Cameras. 2023. Tic peripheral: Search camera head (thermal).

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions.

Water Sampling. 2023. East palestine train derailment information.

Yaqi Xie, Chen Yu, Tongyao Zhu, Jinbin Bai, Ze Gong, and Harold Soh. 2023. Translating natural language to planning goals with large-language models.

Holly Yan, Christina Maxouris, and Nicki Brown. 2023. The ohio toxic train wreck was '100% preventable' – but there's no evidence the crew did anything wrong, investigators say.