Physical Common Sense through Simulation

Eric Robertson
University of Maryland
eric@ericeric.dev

Abstract

As a key ability in the advancement of natural language understanding, language models must be able to reason abstractly about the natural world through physical common sense. In recent year this has become accepted by existing research, but explorations into this domain often focus on commonplace objects and human-annotated examples. Though these approaches are successful with their goals, they are costly to create and fail to leverage the capabilities of purpose built physics engines for modeling the natural world. In this paper, I present a novel approach using computer simulation to generate large scale natural language training sets for physical common sense without human annotation.

1 Background

Previous works have used human annotation to extract relations between physical objects (Forbes and Choi, 2017) and have found success in asking humans to label relative physical properties between objects in extracted text. In the example "She ran into the barn", the annotator could produce results such as "She is smaller than the barn" and "She is faster than the barn", allowing a knowledge graph approach to start with a web of common sense relations effectively placing all objects on different continuous axies of relation. Further research into physical common sense extends to the PIQA (Bisk et al., 2020) dataset which similarly relies on annotators to provide novel edits and solutions to problems based on the physical properties of the in-question objects. In both of these approaches, existing data was used and annotated to provide a physical common sense over the natural world. I pose that the lack of abstract-focused datasets poses a potential issue to advancements of the field and abstract knowledge is key to many of the complex reasoning humans do.

2 Dataset Generation

At a high level, we will be using the physics engine Unity3D to construct a randomized simulation and use features built into the engine to record what events happen as ground truth. From these event records we are able to generate natural language descriptions of both the scene as it was constructed and of the scene as it plays out in simulation. Unity 3D was selected as it is commonly used as a video-game development tool and comes with a large ecosystem of support. In addition to its namesake 3D mode, it supports 2D versions which will be used in this paper for simplicity (See Figure 1). For each simulation, we then make changes pragmatically and record new results with various scripting and trigger based actions in our simulations (which Unity3D refers to as scenes). Using these scripts, a set of standardized properties can be modified on objects such as position, mass, size, velocity and others explored later in this work. It is the ultimate goal to generate a dataset where the abstract idea of "gravity" or of "mass" is able to be captured so models may be able to internally simulate more complex interactions in the real world.

Generation Methods To generate new scenes, we randomly choose settings (a flat platform, slope, staircase, ...), a number of objects (1,2,3, or 4), the type of each object (ball or cube), their relative positions (cube above ball, ball left of cube, ...) and a selection of 14 random attributes per object (bounciness, mass, size, ...). These collection of properties is labeled the "scene description". From it, 10 "scene interpretations" are constructed that meet these constraints but are slight alterations of each other. Minor object position

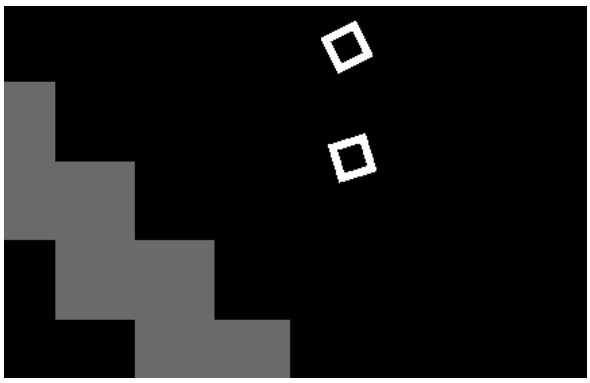

Figure 1: Screenshot of simulation mid-run. A 2D view with simple abstract shapes. In this example, two cubes above a set of stairs

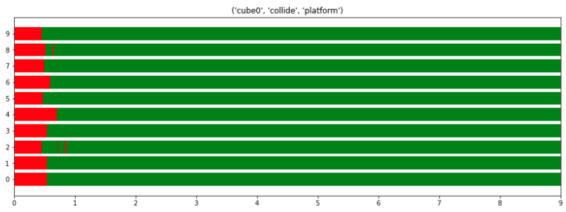

Figure 2: Visualization of timeline for event "Cube0 collides with platform" across the 10 variation the scene. Red indicated false, Green true. The cube is clearly seen to spawn in mid-air above the platform and false down. In two interpretations, it bounces

changes, velocities, rotations, and other properties are tweaked and other properties are modified to provide a means to capture various ways the scene could be pictured by humans and thus remove variability. Properties modified in the scene interpretation are different and (assumed) less impact variables from the scene description.

Collection of Events Events are conditions occurring within a scene including collisions, rotations, translations and others that are captured along with the specific time of the physics simulation since the start of the scene. As each scene interpretation is slightly different from each other, the resultant pattern of events will be similar but divergent, allowing some ability to determine what events are a result of randomness in interpretation and which events are fundamental to the construction of a scene. By selecting a target event across all interpretations of a scene description, a timeline can be generated showing when it was recorded (See Figure 2). Areas of high correlation where all interpretations of the scene were in agreement are taken as key indicators of fundamental events. Areas of low or mixed agreement are thrown out as recordings stemming from randomness. For events that are continuous (ie: two objects in contact with each other) rather than instantaneous (ie: two objects colliding with each other), it is important to record the polarity across ranges. Events that mark the beginning of an action receive a positive score in polarity (+1) whereas events marking the ending are given a negative (-1) score for polarity. Polarity can also have a larger magnitude if across a given range an event is recorded to begin and end rapidly.

Not all timelines have as high-agreement as pictured in Figure 2. Agreement is rather simply measures by the percent of time a given event is true or false across a specific time range (say, from 1 to 3 seconds in the simulation). If, within a given range, the event is always false, or always true, or very nearly one of these extremes, then we can assume high agreement. If there is high polarity on some interpretations but not all, or if one interpretation thinks its purely false while another sees purely true, we see low agreement. Where high disagreement exists, timelines are thrown out as a variable that is subjective in the given scene description.

3 Conversion to natural language

In total from the simulation we have created a representations of the scene description, a list of the events that occurred across the 10 interpretations of a scene, and the agreement scores for the interpretations cross certain predefined ranges of time. Templates are then used to convert the problem into a natural language statement. Taking multiple specified range of simulation time from all interpretations allows multiple questions to be asked about the same event as it changes over time. As a result, a given random scene can spawn hundreds of true statements backed by simulation data. For generation of false statements, the inverse implications can be created with simple string matching and replacing (ie, "certain" being replaced with "impossible").

The scene description is similarly converted through a different set of templates. Together, 44,000 pairs of context and hypothesis sentences were generated and labeled as either true or false. The actual generate step was relatively resource-cheap and given a larger scale version of this exploration, tens of millions of examples could be

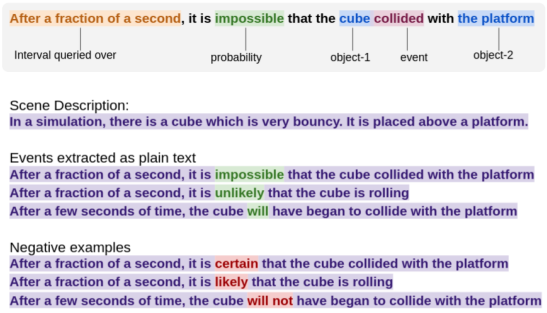

Figure 3: Example sentence construction with templates for timeline, probability, object, event, and object 2

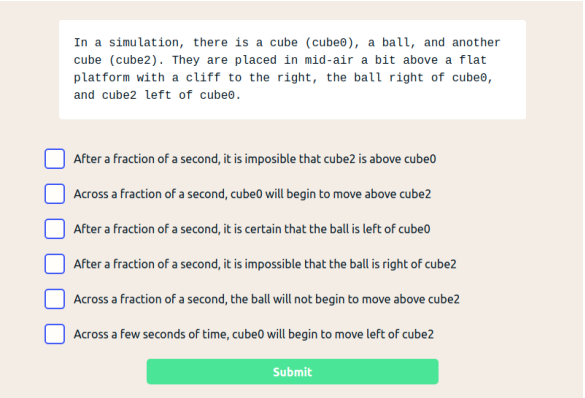

Figure 4: Example of annotation asked to students

reasonably be generated.

4 Annotation

Though the goal is to escape human annotation as the source of data generation, human annotation is a good benchmark for data validity. A website was created and opened up to students for annotation of sample scene-sentence pairs. Across the relatively small number of participants, 44 at the time of writing, over 300 pairs were annotated with a total accuracy of 70 percent. This is rather low in comparison to other human benchmarks, and is likely hurt by a combination of factors. Specifically the way the website is presented (it's not the most user friendly), the overwhelming number of questions thrown at visitors (6 questions is a bit high per page), and the complexity of some scenes (4 objects can be hard to track). As these issues are addressed the average does improve but a larger scale repeat would require more time and resources outside of this project. Regardless, it's value does suggest there is merit in this technique and a larger scale and better explored approach would show higher values.

5 Machine Performance

As a machine benchmark, BERT encodings were used to encode the generated context-hypothesis pairs into a pair of vectors. A simple linear classifier was then able to classify each vector pair as either true or false with more than 95 percent accuracy. This makes me assume there is a good amount of artifacting taking place. Attempts to address this have so-far failed but are likely stemming from the way false examples are generated. As of this writing they are created through simple string matching, but a better approach could entail running simulations that are specifically designed to be contradictory to the scene description generated. Furthermore, the training of a model to include this dataset in its training data was unable to be explored given resources available at the time of writing, but a more thorough experiment could show to what degree this data is beneficial to training a model.

6 Future Research

From the results it is clear that there is promise in machine-generated training sets for physical commonsense reasoning, but such approaches need careful selection of modifiers and generation techniques to avoid confusing humans and introducing artifacts to machines. The scenes constructed were rather basic and used simple geometries with rigid bodies and little actual variations. Using more advanced simulations with more interpretations run for each scene could be just what language models need to open their eyes to the physical world.

The full source for this project is living here: https://github.com/eric-robertson/828-final

References

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7432–7439.

Maxwell Forbes and Yejin Choi. 2017. Verb physics: Relative physical knowledge of actions and objects. *arXiv preprint arXiv:1706.03799*.