EXPLAGRAPHS: An Explanation Graph Generation Task for Structured Commonsense Reasoning

Swarnadeep Saha **Prateek Yadav** **Lisa Bauer** **Mohit Bansal**
UNC Chapel Hill
{swarna, prateek, lbauer6, mbansal}@cs.unc.edu

Abstract

Recent commonsense-reasoning tasks are typically *discriminative* in nature, where a model answers a multiple-choice question for a certain context. Discriminative tasks are limiting because they fail to adequately evaluate the model's ability to reason and explain predictions with underlying commonsense knowledge. They also allow such models to use reasoning shortcuts and not be "right for the right reasons". In this work, we present EXPLAGRAPHS, a new *generative* and *structured* commonsense-reasoning task (and an associated dataset) of explanation graph generation for stance prediction. Specifically, given a belief and an argument, a model has to predict whether the argument supports or counters the belief and also generate a commonsense-augmented graph that serves as non-trivial, complete, and unambiguous explanation for the predicted stance. The explanation graphs for our dataset are collected via crowdsourcing through a novel *Collect-Judge-And-Refine* graph collection framework that improves the graph quality via multiple rounds of verification and refinement. A significant 83% of our graphs contain external commonsense nodes with diverse structures and reasoning depths. We also propose a multi-level evaluation framework that checks for the structural and semantic correctness of the generated graphs and their plausibility with human-written graphs. We experiment with state-of-the-art text generation models like BART and T5 to generate explanation graphs and observe that there is a large gap with human performance, thereby encouraging useful future work for this new commonsense graph-based explanation generation task.[1]

1 Introduction

In the past few years, numerous commonsense reasoning benchmarks have been developed that challenge ML models to use various kinds of commonsense knowledge for solving tasks (Davis and Marcus, 2015). Recent state-of-the-art commonsense reasoning models are typically trained and evaluated on *discriminative* commonsense reasoning datasets and tasks, in which a model answers a multiple-choice question for a certain context (Zellers et al., 2018, 2019; Talmor et al., 2019; Sap et al., 2019b; Bisk et al., 2020). While pre-trained language models perform well on these tasks (Lourie et al., 2021), this setup severely limits the exploration and evaluation of a model's ability to reason and explain its predictions with relevant commonsense knowledge. In fact, neural models are often right for the wrong reasons (McCoy et al., 2019) and use statistical biases or annotation artifacts to solve tasks via shortcuts (Gururangan et al., 2018).

Thus, we emphasize the importance of *generative* commonsense reasoning capability, in which a model is challenged to compose and reveal the plausible commonsense knowledge that is required to solve a reasoning task. Moreover, structured (e.g., graph-based) commonsense explanations, unlike unstructured sentence-based explanations, can more explicitly explain and evaluate the reasoning structures of the model by visually laying out the relevant context and commonsense knowledge edges, chains, and subgraphs. We propose EXPLAGRAPHS, a new *generative* and *structured* commonsense-reasoning task (in English) of explanation graph generation for stance prediction on popular debate topics. Specifically, our task requires a model to predict whether a certain argument supports or counters a belief about a debate topic, but correspondingly, also generate a commonsense explanation graph that explicitly lays out the reasoning process involved in inferring the predicted stance. For example, consider Fig. 1 which shows two examples from our benchmarking dataset EXPLAGRAPHS collected for this task.

[1] EXPLAGRAPHS will be publicly available at https://github.com/swarnaHub/ExplaGraphs.

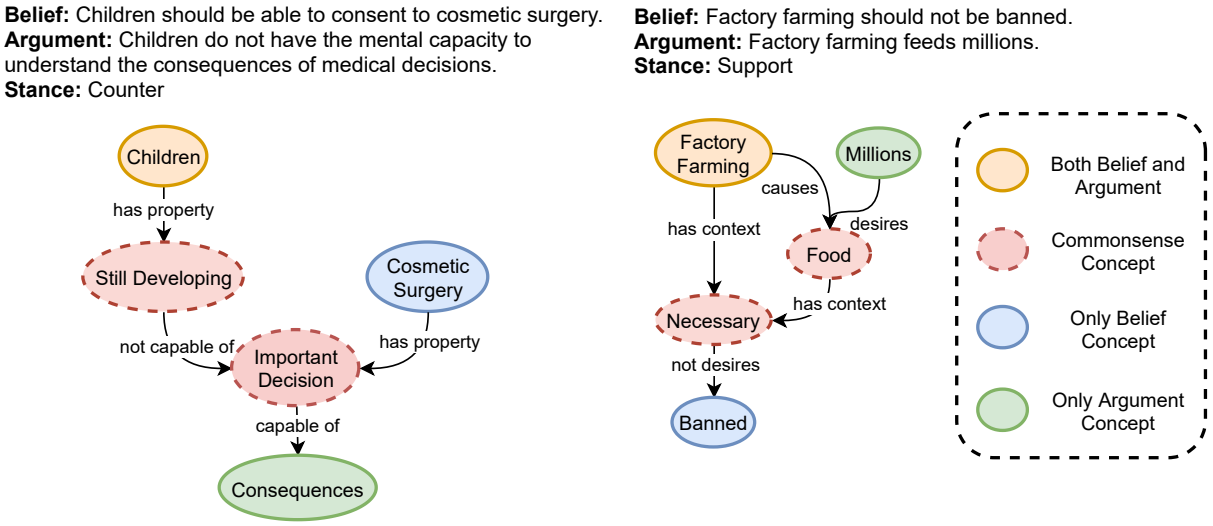


Figure 1: Two representative examples from our dataset showing the belief, the argument, the stance label and the corresponding commonsense explanation graph. The graphs are read by following the edge directions to express the reasoning process involved in explaining why the argument supports or counters the belief.

Each example contains a belief, an argument, and a stance label of either "support" or "counter". Each belief-argument pair requires understanding social, cultural, or taxonomic commonsense knowledge about debate topics in order to infer the correct stance. Specifically, the example on the left requires the knowledge that "children" are "still developing" and that this indicates that they are not capable of making an "important decision" and that "cosmetic surgery" is an "important decision", and that an "important decision" is capable of "consequences". Given this knowledge, one can understand that the argument "children do not have the mental capacity to understand the consequences of medical decisions." is counter to the belief "children should be able to consent to cosmetic surgery". We represent this knowledge in the form of a commonsense explanation graph which allows for causal relationships, ease of imposing constraints, flexibility, and expressiveness. We discuss this and our explanation graph's syntax and semantics below.

Our graph-based explanations follow a broad line of work on structured explanations for NLP. These typically include a chain of facts (Khot et al., 2020; Jhamtani and Clark, 2020; Inoue et al., 2020; Geva et al., 2021) or are semi-structured templates (Ye et al., 2020; Mostafazadeh et al., 2020). As an important next step in this useful line of work, we propose explanations that are fully structured, represented in the form of graphs. Graphs are an efficient data structure for representing explanations because of multiple reasons: (1) unlike chain of facts, they can capture complex dependencies between facts, while also avoiding redundancy (e.g., "Factory farming causes food and millions desire food" forms a "V-structure"), (2) unlike natural language or free-form explanations (Camburu et al., 2018; Rajani et al., 2019; Narang et al., 2020; Brahman et al., 2020; Zhang et al., 2020), it's easier to impose task-specific constraints on graphs (e.g. connectivity, acyclicity), that eventually help in better quality control during data collection (Sec. 4) and designing structural validity metrics for model-evaluation (Sec. 6) and (3) unlike semi-structured templates or extractive rationales (Zaidan et al., 2007; Lei et al., 2016; Yu et al., 2019; DeYoung et al., 2020), they allow for more flexibility and expressiveness (e.g., graphs can encode any reasoning structure and the nodes are not limited to just phrases from the context).

Our explanations specifically take the form of connected directed acyclic graphs (DAG). The nodes in the graph can be concepts (short phrases) from the belief, or the argument, which we refer to as internal nodes. They can also be external commonsense concepts which are neither part of the belief nor the argument but essential in the context for the explanation graph to adhere to the stance. In Fig. 1, these external concept nodes are marked in dashed-red while internal concepts are marked with solid borders. Edges in the graph connect two concepts and are labeled with commonsense relations. The relations are chosen from a pre-defined set and

help form simple coherent facts in conjunction with the two concepts. While some of these facts might not necessarily be factual (e.g. "Factory farming; has context; necessary"),[2] note that such facts are essential in the context for composing an explanation that is indicative of the stance. Semantically, our graphs can be seen as extended structured arguments, augmented with commonsense knowledge.

We construct a benchmarking dataset for our task through a novel framework for collecting graph-structured data via crowdsourcing. Specifically, we propose a *Collect-Judge-And-Refine* graph collection framework, in which we collect connected directed acyclic graphs that serve as non-trivial, complete and unambiguous explanations for the task (as explained in Sec. 6). The framework also allows for iteratively improving the initial graphs through multiple rounds of verification and refinement. A significant 83% of the graphs in our dataset contain external commonsense nodes, indicating that commonsense knowledge is a critical component of our task. The graphs also have reasoning depths of up to 8, suggesting the hardness of the task.

Given that explanation graph generation is a structured prediction task, it poses several syntactic and semantic challenges and consists of multiple sub-problems. First, the model has to generate the nodes which can be further broken down into (a) identifying the internal nodes which are part of the belief and argument, (b) generating external commonsense concepts that are essential for connecting the belief and argument. Second, it has to predict the presence and direction of edges between these concepts and label them with appropriate commonsense relations such that each edge forms a semantically coherent fact. Third, the predicted nodes and edges should lead to a connected DAG. Finally, semantically, the explanation graph should be non-trivial (not paraphrasing the belief as a fact), complete (explicitly connects the argument to the belief) and unambiguously infers the target stance label (Sec. 3).

We also present a comprehensive multi-level evaluation framework for our task, consisting of diverse metrics and human evaluation for our models. The evaluation framework checks for stance and graph consistency along with the structural and semantic correctness of explanation graphs, both locally at the level of each fact (edge) by checking for its importance in improving the model confidence and globally for the whole graph, defined by its ability to reveal the target stance label. Furthermore, we also propose plausibility metrics that match generated graphs with human-written graphs by extending standard text-generation metrics like BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) for graph matching.

Following past work on explanation generation (Rajani et al., 2019), we propose some initial baseline models for our task – (1) a reasoning model that first generates the graph and uses it as additional context with the belief and argument to then predict the stance, and (2) a rationalizing model that first predicts the stance and then generates the graph as post-hoc explanation. Across these models, we represent graphs as linearized strings ordered topologically and fine-tune state-of-the-art pre-trained generative language models BART (Lewis et al., 2019) and T5 (Raffel et al., 2020) on our dataset. Our experiments demonstrate the model's difficulty in generating commonsense-augmented graph explanations for the stance prediction task, leaving a large gap between model and human performance. We encourage researchers to use our benchmark as a way to improve and explore structured commonsense reasoning capabilities of ML models.

Overall, the main contributions of this paper are:

- We propose EXPLAGRAPHS, a new *generative* and *structured* commonsense-reasoning task of explanation graph generation for stance prediction on popular debate topics.
- We release a benchmarking dataset for this task and propose a novel Collect-Judge-And-Refine graph collection framework for collecting graphs that serve as explanations for the task. Our framework is generalizable to any crowdsourced collection of graph-structured data.
- We propose a multi-level evaluation framework for our task, to account for both structural and semantic correctness of graphs and plausibility with human-written graphs.
- We conduct experiments with state-of-the-art text generation models like BART and T5 and find that these models are relatively weak at generating the reasoning graphs, obtaining only 17% accuracy in semantically correct graphs and encouraging useful future work for commonsense graph-based explanation generation.

[2]While some of these beliefs can span controversial debate topics, we do not promote/stand with either the supportive or counter sentiment of these topics and only present both sides of the argument for dataset completeness.

2 Related Work

2.1 Explainable NLP and Structured Explanations

Recent years have seen a surge of interest in explanation datasets for NLP (Wiegreffe and Marasović, 2021), where instances are of the form (input, label, explanation) and are developed with three primary goals: (1) additional supervision to better predict labels, (2) a training signal for generating model explanations, and (3) an evaluation set for comparing model explanations. Explanations in NLP consist broadly of three categories: (1) highlights or extractive rationales (Zaidan et al., 2007; Lei et al., 2016; Yu et al., 2019; DeYoung et al., 2020) which contain subsets of the input (either words, subwords, or full sentences) that are both compact and sufficient to explain a prediction, (2) free-form textual or natural language explanations (Camburu et al., 2018; Rajani et al., 2019; Narang et al., 2020; Brahman et al., 2020; Zhang et al., 2020) which are not constrained to the input and hence are more expressive, and (3) structured explanations (Jansen et al., 2018; Mihaylov et al., 2018; Jhamtani and Clark, 2020; Ye et al., 2020; Inoue et al., 2020). Structured explanations take various forms: explanations graphs (Jansen et al., 2018; Jansen and Ustalov, 2019; Xie et al., 2020), chain of facts or reasoning steps (Khot et al., 2020; Jhamtani and Clark, 2020; Inoue et al., 2020; Geva et al., 2021) and semi-structured text (Ye et al., 2020). Our commonsense explanations are also structured explanations, bearing most similarity to WorldTree's (Jansen et al., 2018) explanation graphs. However, while Jansen et al. (2018) connect words, we connect concepts to create facts and diverse reasoning structures in fully-structured graphs with carefully designed constraints for explainability.

2.2 Commonsense Reasoning Benchmarks

A large variety of commonsense reasoning tasks have been developed in the recent past, including commonsense extraction (Li et al., 2016; Xu et al., 2018), next situation prediction (Zellers et al., 2018, 2019), cultural, social, and physical commonsense understanding (Lin et al., 2018; Sap et al., 2019a,b; Bisk et al., 2020; Hwang et al., 2020; Forbes et al., 2020), pronoun disambiguation (Sakaguchi et al., 2020; Zhang et al., 2020), abductive commonsense reasoning (Bhagavatula et al., 2019) and general commonsense (Talmor et al., 2019; Huang et al., 2019; Wang et al., 2019; Boratko et al., 2020). However, while there is an abundance of discriminative commonsense tasks, there are few recent works in generative commonsense tasks. Most notably, CommonGen (Lin et al., 2020) focused on generating unstructured commonsense text. Instead, our work focuses on generating structured commonsense explanation graphs.

2.3 Stance Prediction and Argumentation

Previous work on stance prediction has been largely applied to online content, often in the domain of political and ideological debates, rumor detection, and fake news detection (Mohammad et al., 2016; Derczynski et al., 2017; Hardalov et al., 2021). Previous work in the space of stance detection, related to popular debate topics and argumentation, has focused on argument convincingness. Gleize et al. (2019) consider pairs of evidence and determine which evidence is more convincing for a claim and Habernal and Gurevych (2016) rank supporting arguments as most to least convincing. Similarly, reason classification for stance prediction of ideological debates has been studied (Hasan and Ng, 2014). However, no previous work requires generative explanations for stance prediction nor explicitly requires commonsense knowledge for their task. To the best of our knowledge, our work is the first to explore explicit commonsense-augmented explanations for the stance prediction task.

3 Task Definition

We propose a new *generative* and *structured* commonsense-reasoning task, where given a certain belief about a topic and an argument, a model has to (1) infer the stance of whether the argument supports or counters the belief, and (2) generate the corresponding commonsense explanation graph that explains the inferred stance (see examples in Fig. 1). The explanation graph $\mathcal{G}$ is a connected and directed acyclic graph (DAG), consisting of a set of nodes $\mathcal{N}$ and edges $\mathcal{E}$. Each node $n_i \in \mathcal{N}$ is a concept, where a concept is defined as an English phrase (often a noun phrase). Concepts can be either internal or external; an internal concept is one which is either part of the belief or the argument while an external concept is one which is neither part of the belief nor the argument but is essential for filling in any knowledge gap between the belief and the argument. Each directed edge $e_i \in \mathcal{E}$ connects two concepts and is labeled with one of the pre-defined set of commonsense relations (see

appendix for the full list of relations).[3] Semantically, our explanation graphs are commonsense-augmented structured arguments that explicitly support or counter the belief. All subjective claims made in the graph are assumed to be true which then is indicative of the stance. An explanation graph is considered correct if it is both structurally and semantically correct.

Structural Correctness of Graphs: In order to ensure the structural validity of a commonsense explanation graph, we define certain constraints on the graph which not only ensure better quality control during our data collection (Section 4) but also simplify the evaluation (Section 6), given the open-ended nature of the task of explanation graph generation. Note that most of these constraints are only possible to impose because of the explicit graphical structure of these explanations. We impose the following constraints.

- Each concept should contain a maximum of three words and each relation should be chosen from the pre-defined set of relations.
- The total number of edges in the graph should be between 3 and 8, to ensure a good balance between under-specified and over-specified explanations.
- The graph should contain at least two concepts from the belief and at least two from the argument. This ensures that the graph uses important parts of the belief and argument (exactly, without paraphrasing) to construct the explanation.
- Finally, the graph should be connected to ensure the presence of explicit reasoning chains that connect the argument to the belief. It should also be acyclic to avoid redundancy or circular explanations. E.g., If a graph already contains the commonsense fact "(vegans; antonym of; meat eaters)", then the fact "(meat eaters; antonym of; vegans)" is redundant and hence prohibited.

Semantic Correctness of Graphs: Beyond the structural constraints, we define the following criteria for an explanation graph to be semantically correct. First, all facts in the graph, individually, should be *semantically coherent*. Second, we require that the graph be *non-trivial*, *complete* and *unambiguous*. We call a graph *non-trivial* if it uses the argument to arrive at the belief and does not use fact(s) which are mere paraphrases of the belief. E.g.: Consider the belief "Factory farming should be banned" for the example on the right in Fig. 1. If the explanation graph contains facts like "(Factory farming; desires; banned)" or "(Factory farming; not desires; banned)", then it is a trivial graph since it is just paraphrasing the belief to explain why the belief does or does not hold. A non-trivial explanation graph should be augmenting the argument with commonsense knowledge like in Fig. 1, which states that "Factory farming causes food and millions desire food which is necessary and hence should not be banned", thereby supporting the belief. A *complete* graph is one which explicitly connects the argument to the belief and no other commonsense knowledge is needed to understand why it supports or counters the belief. For example, in Fig. 1, the fact "(necessary; not desires; banned)" makes the explanation complete by explicitly connecting back to the belief. Finally, we call a graph *unambiguous* if it, as a whole, infers the target stance label and only that label. For example, reading the graph in Fig. 1 in English by following the edges, unambiguously infers the target label. We revisit these definitions of structural and semantic correctness when evaluating the quality of human-written graphs (Section 4.2) as well as model-generated graphs (Section 6).

4 Dataset Collection

We collect EXPLAGRAPHS data via crowdsourcing on Amazon Mechanical Turk (AMT). Figure 2 illustrates our overall collection framework. We separate this crowdsourcing task into two stages. In Stage 1 (illustrated on the left of Fig. 2), we collect instances of belief, argument and their corresponding stance (support or counter). In Stage 2 (illustrated on the right of Fig. 2), we collect the corresponding commonsense explanation graph for each (belief, argument, stance) sample.

4.1 Stage 1: (Belief, Argument, Stance) Collection

In Stage 1, annotators are given prompts that express beliefs about various debate topics. We extract these prompts from evidences in Gretz et al. (2019), balancing the prompts across topics. An example of a prompt is "A 1999 meta-analysis of five studies comparing vegetarian and non-vegetarian mortality rates in Western countries found a 6 percent reduction in mortality from ischemic heart

[3] Since an edge forms a meaningful sentence by combining the two concepts and the relation, we will use edges and facts interchangeably. Similarly, concepts and nodes mean the same in our setting.

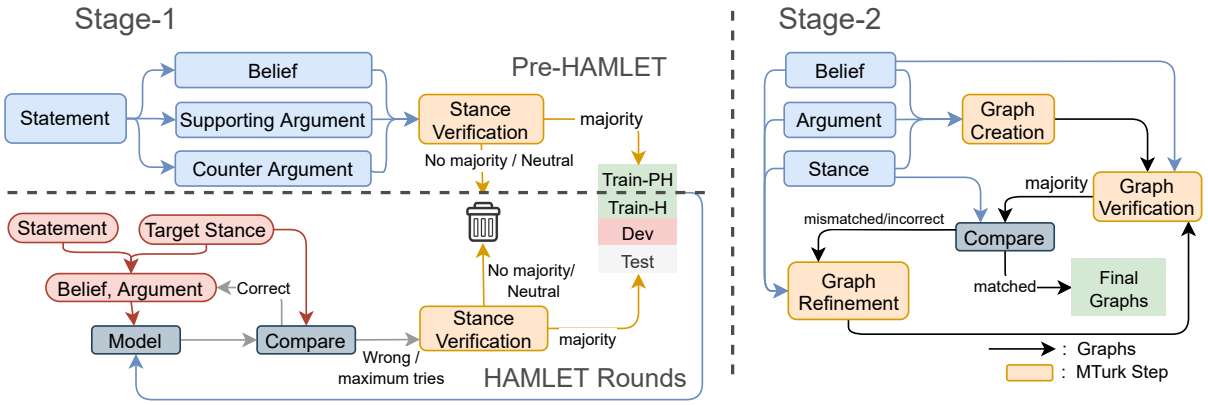


Figure 2: Interface for our data collection framework consisting of two stages. In Stage 1, we collect (belief, argument, stance) triples in pre-HAMLET and multiple HAMLET (human-and-model-in-the-loop) rounds. In each HAMLET round, we collect harder examples by asking the annotators to fool an initial stance prediction model, which then gets improved in each successive round with data from the previous rounds. In Stage 2, we collect the corresponding commonsense explanation graphs through a Collect-Judge-and-Refine framework.

disease in vegans compared to occasional meat eaters." that expresses the belief that "Vegans can live longer than meat eaters". We use 71 topics in total during our data collection process, randomly assigning 53/9/9 topics to our train/dev/test data splits respectively, ensuring disjoint topics across splits. The appendix contains the list of all topics.

Given the prompt, annotators are asked to write the belief expressed in the prompt and subsequently write a supporting argument and a counter argument for the belief. The beliefs and arguments are typically one-sentence long, containing a maximum of 30 words. Since we focus on explanations augmented with commonsense knowledge, we want to ensure that most of our collected belief, argument pairs are non-trivial and require some implicit background commonsense knowledge for understanding why a certain argument supports or refutes the belief. Consider the right example in Fig. 1, where the explanation graph lays out implicit world and commonsense knowledge like "(factory farming; causes; food)" and "(food; has context; necessary)" which are neither specified in the belief nor in the argument but necessary for understanding how the argument supports the belief. In order to collect such (belief, argument) pairs, we use Human-And-Model-in-the-Loop Enabled Training (HAMLET) (Nie et al., 2019), a multi-round adversarial data collection procedure that enables the collection of trickier examples with more background commonsense knowledge (as described below). Thus, we further split this stage into two parts: pre-HAMLET collection (top part of left of Fig. 2) and HAMLET collection (bottom part of left of Fig. 2). During pre-HAMLET collection, we collect some initial belief, argument pairs for training a state-of-the-art model for stance prediction, which then facilitates the collection of harder examples in multiple HAMLET rounds.

Pre-HAMLET: The complete instructions for pre-HAMLET data collection is in the appendix. Briefly, annotators write the belief expressed in the prompt along with a supporting and a counter argument. We collect a total of 998 samples from randomly chosen 33 topics out of the 53 train topics with an average of 30 samples per topic. Note that we do not include the dev and test topics as part of the pre-HAMLET collection to ensure that the examples in these splits are sufficiently hard for the models.

HAMLET: We follow the initial pre-HAMLET collection round with 3 rounds of HAMLET collection to reduce any annotation artifacts and most importantly, collect harder examples with implicit background knowledge. At each round of HAMLET collection, we ask annotators to write (belief, argument) pairs in a way that a stance prediction model is fooled. In the first round, we start by fine-tuning a RoBERTa model (Liu et al., 2019) on the pre-HAMLET data that given a (belief, argument) pair predicts the stance label. After each round, we divide the collected HAMLET data into train, dev and test splits based on their respective topics and update the RoBERTa model by training on the pre-HAMLET data and the train splits of the HAMLET

rounds collected so far. We collect data in each round from the remaining 38 topics (20 train, 9 dev, 9 test) equally. In contrast to the pre-HAMLET round, here we also provide the target stance label along with the prompt and annotators are asked to write the belief and an argument that adhere to the target label. Once they construct a pair, in real-time, it is sent to the stance prediction model and if the model is able to predict the stance correctly, we prompt the annotators to rewrite either the belief or the argument. We provide annotators 3 tries in Round 1 and 4 tries in Round 2 and Round 3 to fool the model, following which we accept the final pair. Our HAMLET collection comprises of a total of 2170 samples with 892, 667 and 611 samples in rounds 1, 2, and 3 respectively.

Quality Control for Stage1: We apply the following mechanisms to control the quality of the collected data.

- **Onboarding Test:** Each annotator is required to successfully pass an onboarding quiz before they can start writing belief and argument pairs. In this test, we evaluate their understanding of supportive and counter arguments by providing them with 10 (belief, argument) pairs and they are asked to choose if the argument supports or counters the belief.
- **Stance Label Verification:** We verify the stance labels of all the examples collected in pre-HAMLET and HAMLET rounds. This is particularly necessary for the HAMLET rounds where the annotators are constrained to fool the model and it is hard to create such samples and hence verification is required. For each (belief, argument) pair, we ask five annotators to choose the correct label between "support", "counter", and "neutral". We choose the majority label as the final label and keep only those examples that have majority labels either "support" or "counter". This results in a high fleiss-kappa inter-annotator agreement score of 0.61. Instructions and interface for the verification task are in the appendix.

4.2 Stage 2: Commonsense Explanation Graph Collection

Given the (belief, argument, stance) triples from Stage 1, we next collect commonsense explanation graphs in Stage 2 that explain why the argument supports or counters the belief (see right of Fig. 2). We introduce a generic *Collect-Judge-And-Refine* iterative framework for collecting high-quality graphs through crowdsourcing. We describe each of these stages in detail below.

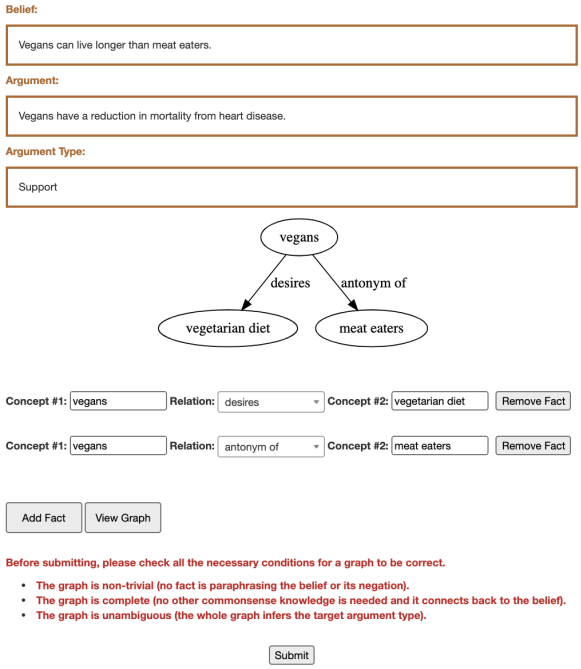


Figure 3: Interface for Commonsense Explanation Graph Collection. Annotators are provided with the belief, argument and the stance label. They construct the commonsense explanation graph by writing multiple facts, each consisting of two concepts and the appropriately chosen relation between them. Clicking on "View Graph" button shows the graph constructed so far.

Graph Collection: Annotators are given a belief, an argument, and the corresponding stance label (support or counter) and are then asked to construct an explanation graph, augmented with background commonsense knowledge and explaining the target stance label. Our interface is shown in Figure 3. A graph is constructed by writing multiple facts, each consisting of two concepts (either internal or external) and a chosen relation that connects the two concepts. They are constrained to write 3-8 facts in a way that the facts lead to a connected and directed acyclic graph with at least two concepts (nodes) from the belief and two from the argument. The graphical representation of the explanation provides an explicit structure, thereby allowing us to automatically perform in-browser checks for these structural constraints.

Our interface also provides a "View Graph" button, clicking which shows the graphical representation of the facts written so far. While there is

no automatic way to check for the semantic correctness of these graphs, before submitting, we remind the annotators that they read and reason through their constructed graph and verify that it is non-trivial, complete and unambiguous (see the red marked text in Figure 3). The appendix contains screenshots of the detailed instructions for graph collection.

Graph Verification: In this stage, we only check for the semantic correctness of graphs (as defined, previously in Sec. 3) because by construction, they are all structurally correct. The explanation graphs are required to be complete and hence are treated as extended structured arguments, augmented with commonsense knowledge. Thus, in our graph verification step, we provide annotators with only the belief and the corresponding explanation graph and ask them to reason through it to infer the stance label (support/counter). Additionally, we include a third category of incorrect graphs which is broadly aimed at identifying the ill-formed graphs with either (1) semantically incoherent facts, (2) facts paraphrasing the belief or its negation (trivial graphs), or (3) no explicit connection back to the belief and hence incomplete or ambiguous. Each graph is annotated by three verifiers into one of the support/counter/incorrect categories. A graph is considered correct if and only if the majority label matches the original stance label (already known from Stage 1). All other graphs are sent to the graph refinement stage (described next, also see Fig. 2) because they are either incorrect or infer the wrong stance label. See appendix for the detailed instructions and interface of graph verification.

Graph Refinement: During the graph refinement stage, in addition to the belief, argument and the target stance label, annotators are provided with the initial incorrect graph along with the majority verification label from the verification stage. Then another qualified annotator who is not the author of the initial graph is asked to refine it. The verification label provides additional signal with respect to whether the graph contains incoherent facts or if the facts are individually correct but the graph infers the wrong stance label. Refinement is defined in terms of three edit operations on the graph: (1) adding a new fact, (2) removing an existing fact, and (3) replacing an existing fact. Our refinement interface is shown in Figure 4. Similar to the collection phase, we ensure that the refined graph

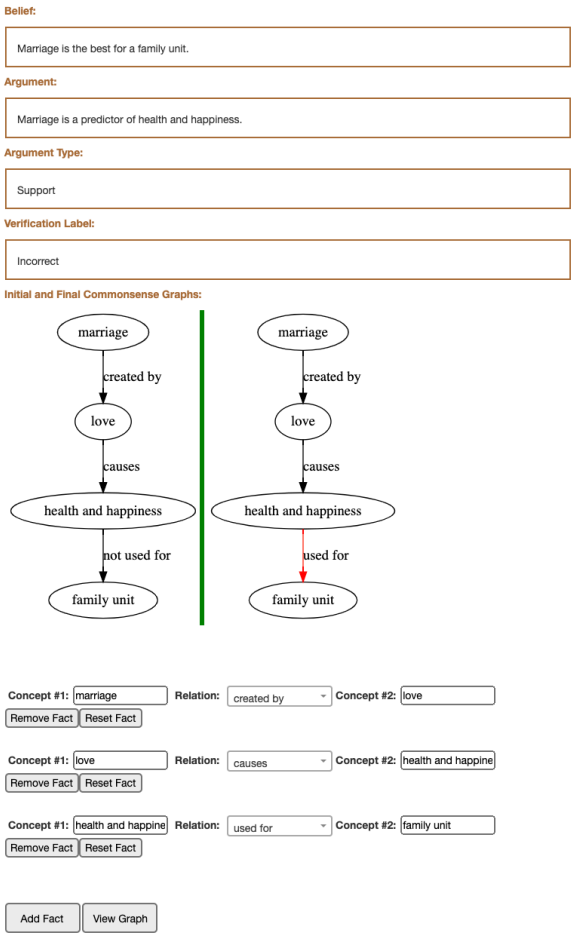


Figure 4: Interface for commonsense explanation graph refinement: Annotators are provided with the belief, argument, the stance label, the initial incorrect explanation graph and the majority verification label. They refine the graph by adding, removing or replacing facts and the changes to the initial graph are shown in red.

also adheres to the structural constraints. The view graph button shows the updated graph, with the changes marked in red. See appendix for the detailed instructions of explanation graph refinement.

The refined graphs are again sent to the verification stage and the process iterates between these two stages of verification and refinement until we obtain a high percentage of correct graphs. Thus far, we have performed two rounds of verification and one round of refinement, and we observe a high 81% of semantically correct graphs (see Table 2). Without any refinement, we started off with 63% correct graphs, which further increased to 81% after one round of refinement. We note that our *Collect-Judge-And-Refine* framework is generic and can be repeated any number of times to in-

Round	Train S / C	Train Total	Train Topics	Dev S / C	Dev Total	Dev Topics	Test S / C	Test Total	Test Topics
Pre-HAMLET	541 / 457	998	33	-	-	-	-	-	-
HAMLET R1	347 / 226	573	20	79 / 76	155	9	84 / 80	164	9
HAMLET R2	234 / 181	415	20	66 / 63	129	9	64 / 59	123	9
HAMLET R3	213 / 169	382	20	55 / 61	116	9	52 / 61	113	9
EXPLAGRAPHS	1335 / 1033	2368	53	200 / 200	400	9	200 / 200	400	9

Table 1: EXPLAGRAPHS dataset statistics: S = Support, C = Counter, Topics = Number of spanning topics.

	Correct	Wrong Label S → C	Wrong Label C→S	Incorrect
Round 1	63	0	8	29
Round 2	81	0	1	18

Table 2: Overall graph correctness across rounds (in %). S→C denotes the percentage of samples where the majority label from the verification stage is counter instead of support (actual label), similarly for C→S. Incorrect refers to the percentage of graphs that have been marked incorrect. Note that both wrong label and incorrect graphs are sent for refinement.

crease the quality of the dataset.[4] More broadly, our graph collection and refinement stages can be easily adapted towards collecting more graph-related datasets in the future.

Quality Control for Stage2: Quality control of crowdsourced data is challenging, more so when the task involves creating graphs with associated constraints like acyclicity, connectivity, etc and then reasoning through the graph to infer the target label. Verifying these graphs for completeness, semantic coherence and non-triviality also requires understanding the overall motivation of the underlying task and hence is significantly more challenging than our Stage 1 stance label verification. In the light of these challenges, we employ carefully designed quality control mechanisms, which we believe will be helpful for similar graph collection tasks in the future:

- **2-level Onboarding Test:** Since the three stages of graph creation, verification and refinement are closely tied to one another, we choose a single pool of annotators to perform all the graph-related tasks. We also prohibit annotators from verifying their own graphs. We design a 2-level onboarding test where in the first level, we test the annotators' understanding of a commonsense fact because that is the basic building block of our graphs. Annotators are tested on 10 multiple choice questions, half of which require choosing the correct relation given the two concepts and another half require choosing the right pair of concepts, given the relation. Successful annotators from the first level qualify for the second level, where they are required to take two other tests. In one, we ask them to create a graph given a (belief, argument, stance) triple, whose quality we manually verify and in another, we ask them to verify the correctness of some already provided explanation graphs. See appendix for the detailed instructions of this onboarding test.
- **Intensive Training and Feedback:** We begin by providing detailed feedback and explanations of the correct answers from the onboarding tests to every qualified annotator. Every new annotator who starts creating graphs for the first time is initially requested to submit only a small number of graphs. We then verify these graphs manually and provide detailed feedback and suggest improvements wherever there are some incoherent facts in the graph or the graph is a trivial explanation or is incomplete. Over time, we find such personal feedback to be highly effective towards improving the quality of the graphs.
- **High-performing annotators for Refinement:** While it is theoretically possible to run multiple iterations of graph verification and refinement, under most practical scenarios due to time and budget constraints, we want to ensure that one round of refinement is enough to obtain a high percentage of correct graphs. Hence, we qualify only the high-performing annotators (whose graphs have been verified as correct the most) for our refinement task.

5 Dataset Analysis

Dataset Size: EXPLAGRAPHS consists of a total of 3168 examples, with 2368 examples for training,

[4] We are currently running another round of refinement to improve dataset quality further to 90%.

400 for validation and 400 for testing.[5] Table 1 shows the number of samples, distribution of stance labels and the number of spanning topics across multiple rounds of collection. As noted earlier, by design, the topics across the data splits are disjoint and dev and test splits contain examples from the 3 rounds of HAMLET only.

Overall Graph Statistics: Table 3 shows statistics concerning the number of nodes, edges, and external nodes (concepts not part of either the belief or the argument) present in our graphs. Approximately 83% of our samples contain external nodes, indicating that most of our samples require some kind of implicit background commonsense knowledge to explicitly support or refute a belief. Additionally, we see that our graphs have diverse reasoning structures, demonstrated by a significant 56% of graphs with non-linear structures. A linear reasoning structure is one where the nodes in the graph are connected by a single linear chain. A large presence of non-linear structures add to the challenging nature of our task and demonstrate that commonsense explanation requires complex reasoning abilities. We also compute the reasoning depth of our graphs, as defined by the maximal length path between a root and a leaf node in the DAG.

Graph Relation Distribution: The relations used to construct the facts in our graphs can be divided into two categories – with and without negations ("not capable of" vs "capable of"). We analyze the presence of these relations separately for the support and counter graphs. Fig. 5 illustrates that while non-negated relations are used more frequently in both kinds of graphs, they broadly follow a similar distribution of negated vs non-negated relations, demonstrating that the usage of a type of relation is not indicative of the stance label and actually depends on the specific context they are being used in. Interestingly, we also observe that the most frequently used relations in both stances are causal in nature (like "capable of", "causes", "desires", and their negative counterparts), which further supports our graphs as explanations.

[5]While we are continuing our efforts to expand the dataset size further, we note that EXPLAGRAPHS's size is bottlenecked by Stage2, i.e., the graph collection phase. We found that it is significantly challenging to collect complex structured datasets, as also noted by previous works (Geva et al., 2021). Specifically, for collecting graphs, the challenges include training annotators about what graphs (with cycles, connectivity) mean and how to verify them for semantic consistency and stance inference.

6 Evaluation Metrics

Based on our task definition in Section 3, we design evaluation metrics that evaluate the structural correctness as well as the semantic correctness of explanation graphs (refer to Sec. 3). While we do propose plausibility metrics that try to match predicted explanation graphs to human-written graphs, we conclude that such metrics, in line with prior natural language explanation studies (Camburu et al., 2018; Marasović et al., 2020), are not ideal. Explanation graphs can be represented in multiple correct ways with varying levels of specificity. For example, certain explanations can be over-specified, while others can be under-specified leading to different graphical structures and a single concept can also be paraphrased in multiple different ways. Thus, we design an evaluation pipeline, consisting of the following three levels.

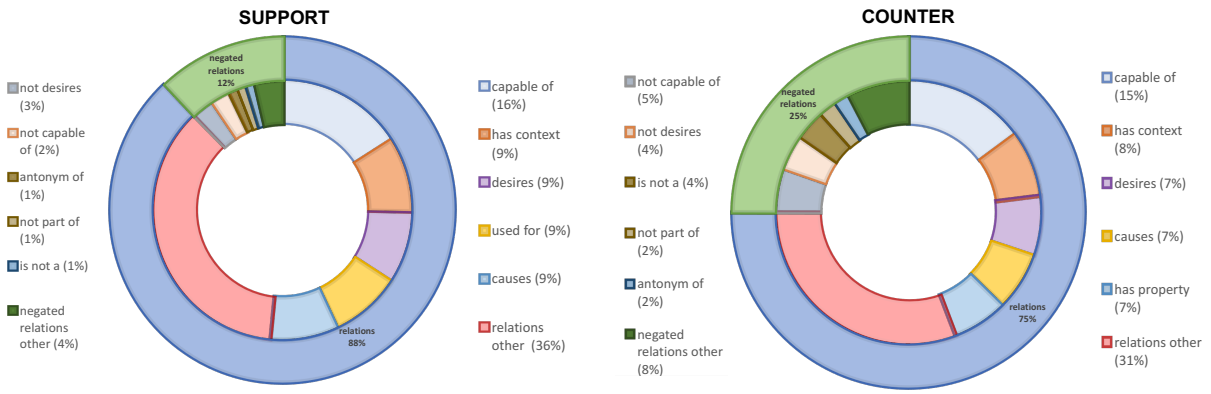
Level 1 – Stance Accuracy (SA): The models for our task are required to predict the stance label along with the corresponding commonsense explanation graph. In Level 1, we report the stance prediction accuracy. Stance correctness is necessary to ensure that the explanation graph is consistent with the predicted stance label. Samples with correctly predicted stance are then passed to the next level of metrics which check for the quality of the generated explanation graphs.

Level 2 – Structural Correctness Accuracy of Graphs (StCA): As per our task definition in Section 3, for an explanation graph to be correct, we first require that it is structurally correct because if the graphs are not structurally correct then we cannot reason over them unambiguously. Thus, we report a single accuracy metric which computes the fraction of structurally correct graphs (connected DAGs with at least three edges and at least two concepts from the belief and at least two from the argument). Note that this fraction is with respect to the size of the entire evaluation set. Samples with correctly predicted stances and structurally correct graphs, are then evaluated in Level 3 along three different parallel axes: (1) semantic correctness, (2) plausibility with human-written graphs, and (3) edge importance.

Level 3 - Semantic Correctness Accuracy of Graphs (SeCA): For evaluating the semantic correctness of explanation graphs, we introduce

	Nodes Min/Max/Mean	Edges Min/Max/Mean	External Nodes Min/Max/Mean	Depth Min/Max/Mean	%Non-Linear	%Ext. Nodes
Train	4/9/5.1	3/8/4.2	0/6/1.3	1/8/3.3	58.72	79.77
Dev	4/9/5.5	3/8/4.6	0/6/1.7	2/8/3.8	48.74	92.23
Test	4/9/5.5	3/8/4.6	0/6/1.6	1/8/3.7	53.13	90.00
All	4/9/5.2	3/8/4.3	0/6/1.4	1/8/3.4	56.76	82.63

Table 3: Gold Graph Statistics: Node represents number of concepts in the graph, External Nodes show the number commonsense concepts that are added to the graphs which are not part of the belief or the argument. Depth denotes the maximal length path between a root and a leaf node. Non-Linear denotes the Percentage of graphs which are not linear chains. % Ext. Nodes denotes the total percent of new commonsense concepts that are introduced in the graphs.



(a) Relation Distribution for Support Graphs

(b) Relation Distribution for Counter Graphs

Figure 5: Relation percentages for Gold Graphs: Frequencies of occurrence of positive and negative relation for support and counter graph along with sub-classification into relation level statistics.

a model-based metric that replicates human verification of graphs discussed in Section 4.2. Since identifying semantic correctness of graphs is challenging and expensive in terms of human effort and money, we propose an automatic model-based metric to do so, following previous works (Zhang* et al., 2020; Sellam et al., 2020; Pruthi et al., 2020). Specifically, we fine-tune a RoBERTa model (Liu et al., 2019) to predict the label (incorrect/support/counter) from the predicted graph and report an accuracy metric – the fraction of graphs for which the predicted label matches the gold label. Note that following our human verification for semantic correctness of graphs, the input to our model is only the belief and the corresponding explanation graph. From our definition of semantic correctness, a graph is classified as *incorrect* if one or more constituent facts are semantically incoherent or if the graph is trivial, incomplete or ambiguous. We automatically obtain training data for this metric model through our human verification stage. For simplicity, graphs are considered as concatenated facts. We train our model on the human-verified graphs of our train split and obtain a reasonable 75% accuracy on the validation split. A key assumption of this metric is that the training data for the incorrect class is representative of all the different ways through which a graph can be semantically incorrect. While we cannot guarantee that, we note that a model-based metric can always be improved in two different ways – (1) by data augmentation through collecting and verifying more graphs or through synthetically created incorrect graphs, and (2) by designing better models. Overall, our semantic-correctness evaluation metric should be seen as an initial attempt for a challenging problem and we encourage future work on developing better metric for the understanding semantic correctness of explanation graphs.

Level3 – Plausibility w.r.t. Human-written Graphs: We also introduce a set of metrics that quantify the degree of match between the human-written graphs and the predicted graphs. Graphs are treated as a set of facts (edges). We solve a matching problem that finds the best assignment between the set of facts in the gold graph and those in the predicted graph. For this, we first define a

	SA	StCA	SeCA	G-BLEU			G-ROUGE-2			G-ROUGE-L			EA
				P	R	F1	P	R	F1	P	R	F1	
Rationalizing-BART	**82.0**	16.0	7.7	3.8	2.8	3.1	4.8	3.5	3.8	7.6	5.7	6.3	5.8
Reasoning-BART	72.2	20.7	13.2	4.7	3.4	3.8	5.9	4.3	4.9	9.6	6.9	7.9	8.2
Rationalizing-T5	**82.0**	**32.7**	13.7	7.0	**5.8**	**6.2**	**9.2**	**7.6**	**8.1**	**15.4**	**12.7**	**13.7**	**13.8**
Reasoning-T5	69.2	32.5	**17.5**	**7.1**	5.6	**6.2**	9.1	7.4	8.0	15.1	12.2	13.3	13.0

Table 4: Table comparing the results of our Rationalizing and Reasoning models with BART and T5 variants across all metrics on EXPLAGRAPHS test set. Vertical lines denote the different levels of our evaluation framework. SA = Stance Accuracy. StCA = Structurally Correct Graph Accuracy. SeCA = Semantically Correct Graph Accuracy. EA = Edge Importance Accuracy. The best numbers are bolded and the second-best numbers are underlined. While all models perform reasonably well on the stance prediction sub-task, they perform poorly on the graph-level metrics. T5-Reasoning model obtains the best accuracy on semantically correct graphs with 17%.

	SA	StCA	SeCA	G-BLEU			G-ROUGE-2			G-ROUGE-L			EA
				P	R	F1	P	R	F1	P	R	F1	
Rationalizing-BART	**84.7**	17.8	10.0	4.1	3.0	3.4	5.2	3.7	4.2	8.3	6.0	6.8	6.8
Reasoning-BART	70.4	18.8	12.5	4.3	3.3	3.7	5.5	4.2	4.7	8.9	6.6	7.4	7.7
Rationalizing-T5	**84.7**	**30.3**	15.0	**6.6**	**5.3**	**5.8**	**8.2**	**6.6**	**7.2**	**13.7**	**10.9**	**11.9**	**11.8**
Reasoning-T5	70.4	27.8	**17.2**	6.1	5.0	5.4	7.8	6.3	6.8	12.7	10.4	11.2	11.5

Table 5: Table comparing the results of our Rationalizing and Reasoning models with BART and T5 variants across all metrics on EXPLAGRAPHS dev set. Vertical lines denote the different levels of our evaluation framework. SA = Stance Accuracy. StCA = Structurally Correct Graph Accuracy. SeCA = Semantically Correct Graph Accuracy. EA = Edge Importance Accuracy. The best numbers are bolded and the second-best numbers are underlined.

scoring function between a pair of gold fact and predicted fact. We treat each fact as a sentence by combining the two concepts and the relation in order and use n-gram matching based metrics like BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) to score a pair of facts.[6] Given the best assignment and the overall matching score, precision is computed as the matching score upon the number of edges in the predicted graph, while recall is computed as the matching score upon the number of edges in the gold graph. Henceforth, we will refer to these metrics as G-BLEU and G-ROUGE.

Level3 - Edge Importance Accuracy (EA): While our graph semantic correctness metric assesses the correctness of a graph at a global level, we also propose a local model-based metric, named "Edge Importance Accuracy" which computes the macro-average of important edges in the predicted graphs. An edge is defined as important if not having it as part of the graph causes a decrease in the model's confidence for the target stance. More specifically, we first fine-tune a RoBERTa model which given a *(belief, argument, graph)*, predicts the probability of the target stance. Next, we remove one edge at a time from the corresponding graph and query the same model with the belief, argument and the graph but with the edge removed. If we observe a drop in the model's confidence for the target stance, we conclude that the edge is useful in indicating the target stance. We first compute the percentage of "important edges" within samples and then average those values out across all samples. Note that this metric like the previous two metrics is also part of our Level 3; hence all samples with either incorrect stance or structurally incorrect graphs obtain no credit against this metric.

[6]The scoring function can also be an embedding-based text similarity metric like BERTScore (Zhang* et al., 2020) or BLEURT (Sellam et al., 2020), but we leave the exploration of such variants as part of future work.

7 Models

We present some initial baseline models that we experimented with for our task. As noted earlier, all our models predict the stance label as well as the corresponding explanation graph. We represent and predict graphs as linearized strings formed by concatenating the constituent edges. Since our explanation graphs are connected DAGs, the edges are concatenated according to the topological order of the nodes. Topological ordering is a natural choice (as opposed to a random ordering) not only because it lays out the order in which a human would read and reason through the edges in the explanation graph but it also helps the model learn

	SA	StCA	SeCA	G-BLEU			G-ROUGE-2			G-ROUGE-L			EA
				P	R	F1	P	R	F1	P	R	F1	
Random	69.2	6.5	3.2	1.3	1.2	1.2	1.8	1.5	1.6	2.9	2.7	2.7	3.0
Topological	**69.2**	**32.5**	**17.5**	**7.1**	**5.6**	**6.2**	**9.1**	**7.4**	**8.0**	**15.1**	**12.2**	**13.3**	**13.0**

Table 6: Effect of edge ordering on Reasoning-T5 model. Having a random ordering leads to a significant drop in performance.

	SA	StCA	SeCA	G-BLEU			G-ROUGE-2			G-ROUGE-L			EA
				P	R	F1	P	R	F1	P	R	F1	
High (>5)	66.6	28.9	1.2	5.8	2.8	3.7	8.1	4.0	5.3	12.9	6.4	8.4	8.5
Medium (4-5)	**71.9**	25.6	5.7	5.2	3.6	4.2	7.3	5.1	6.0	11.8	8.4	9.7	9.7
Low (1-3)	70.0	**34.2**	**8.7**	**6.8**	**6.4**	**6.5**	**9.6**	**9.0**	**9.2**	**15.5**	**14.6**	**14.9**	**14.6**

Table 7: Comparison of Reasoning-T5 model on the subset of examples in EXPLAGRAPHS dev set with varying reasoning depths (low, medium, high). Performance on the graph-related metrics drop significantly with increasing depth.

an inductive bias towards generating graphs in that particular order. Following prior work on explanation generation for NLP tasks (Rajani et al., 2019), we propose the following models.

Reasoning Model (First-Graph-Then-Stance): Our first approach is through a reasoning model which first predicts the explanation graph by conditioning on the belief and the argument and then uses the generated graph, augmented with the belief and the argument, to predict the stance label. The explanation graph, in this case, provides additional commonsense knowledge and structure for the stance prediction task. For the graph prediction model, we fine-tune BART (Lewis et al., 2019) and T5 (Raffel et al., 2020) as two variants, where the input is the concatenated belief, argument and the output is the explanation graph (in topological edge order). Next, for the stance prediction model, we fine-tune a pre-trained sequence classification model, RoBERTa (Liu et al., 2019), which conditions on the concatenated belief, argument and the linearized graph to predict the stance label.[7]

Rationalizing Model (First-Stance-Then-Graph): Our second model is a rationalizing model which generates graphs as post-hoc explanations. Specifically, we first fine-tune a RoBERTa model to predict the stance label by conditioning on the belief and argument. The predicted labels are then concatenated with the belief and argument to fine-tune BART and T5 models for generating the explanation graph in a post-hoc manner.

8 Experiments and Analysis

8.1 Comparison of Rationalizing and Reasoning Models with BART and T5

In Table 4, we compare the performance of Rationalizing and Reasoning models with BART and T5 across all our metrics. We find that in general, T5 generates graphs better that BART, independent of rationalizing or reasoning. Reasoning-T5 and Rationalizing-T5 have comparable performance across all our graph-related metrics. Overall, all our models achieve a significantly low percentage of semantically correct graphs, with Reasoning-T5 getting only 17% of samples with both stance and the corresponding explanation graph correct. This is further reflected in the drop in stance accuracy for the reasoning models in which the stance prediction is conditioned on the generated graph. Finally, the edge importance accuracy is also significantly low for these models, suggesting that the edges generated as part of the graphs do not make the models any more confident about their initial predictions. The failure of these models to correctly generate commonsense explanation graphs is indicative of the challenging nature of our task. Given the large gap between a high 83% of semantically correct graphs in our dataset and the relatively low model performance, we hope our dataset will encourage future work on better model development for ex-

[7]The stance prediction model can possibly be improved with better encoding of the explanation graph (e.g., through graph neural networks). Another interesting line of work could explore the effect of augmentation of external commonsense knowledge sources like ConceptNet (Liu and Singh, 2004) or Atomic (Sap et al., 2019a) for model improvement. We hope our challenging dataset encourages such advanced model development as part of the future work by the community.

	SA	StCA	SeCA	G-BLEU			G-ROUGE-2			G-ROUGE-L			EA
				P	R	F1	P	R	F1	P	R	F1	
Linear	**76.2**	**32.4**	**10.0**	**6.0**	4.7	5.1	**8.6**	6.7	7.4	**14.0**	**11.1**	**12.1**	**12.0**
Non-linear	63.3	27.2	5.25	**6.0**	**5.1**	**5.4**	8.4	**7.0**	**7.5**	13.3	**11.1**	11.9	11.6

Table 8: Comparison of Reasoning-T5 model on the subset of examples in EXPLAGRAPHS dev set with linear vs non-linear graph structures. The stance prediction accuracy (SA) and the structural correctness accuracy (StCA) of graphs drop significantly for non-linear graphs due to the complex reasoning process involved in such graphs.

	Nodes Min/Max/Mean	**Edges** Min/Max/Mean	**External Nodes** Min/Max/Mean	**Depth** Min/Max/Mean	**%Non-Linear**	**%Ext. Nodes**
R-T5	4/6/4.3	3/5/3.3	0/2/0.3	2/5/3.3	10.00	27.91

Table 9: Statistics for the explanation graphs generated by the Reasoning-T5 (R-T5) model. Compared to our gold graphs, the graphs generated by the model have significantly smaller number of external commonsense nodes and lower depths of reasoning. Additionally, most graphs generated are linear chains and only a small fraction of them contain external commonsense concepts.

planation graph generation.[8] Owing to the superior performance of the Reasoning-T5 model, we perform other analysis and ablations with it.

8.2 Effect of Edge Ordering

In order to evaluate the effect of a particular edge ordering on the model's performance, we compare the topologically ordered Reasoning-T5 model with a random ordering model in which the edges are shuffled in any arbitrary order. From Table 6, we observe that having a predefined ordering helps the model to learn the graph structure significantly better. This, however, is not surprising because due to the autoregressive nature of these text generation models, an unordered edge set confuses the model and it is not able to learn the structural properties of graphs. We observe that the random model often generates cycles and hence has a significantly low percentage of structurally correct graphs. Having a fixed ordering (topological in our case) also enables the model to learn an inductive bias towards generating graphs in a manner than can be read and reasoned through by humans.

8.3 Analysis with Reasoning Depths

Since our graphs are connected DAGs, we refer to the depth of the graph as the reasoning depth involved in inferring the stance label. As part of ablation analysis, in Table 7, we analyze the performance of the Reasoning-T5 model on the subset of examples requiring varying depths of reasoning from low (depth $<= 3$) to high (depth > 5). Unsuprisingly, we find that our task of explanation graph generation becomes challenging with increasing depth, as demonstrated by a linear drop in all graph-related metrics from low to medium to high. This again reveals the hardness of our task and encourages future work on better model development of explanation graph generation.

8.4 Analysis with Reasoning Structures

Our next ablation analyzes the effect of linear vs non-linear reasoning structures. We call a reasoning structure linear when the explanation graph contains a single chain of nodes. A non-linear reasoning structure adds complexity to the inference process and we validate this through our results in Table 8. Similar to the previous result, we observe that our task becomes challenging with non-linear structures as demonstrated by a significant drop in both stance accuracy and structurally correct graph accuracy.

8.5 Human Evaluation

While we develop a comprehensive automatic evaluation framework, evaluating graphs for semantic correctness is a challenging problem and we have only explored some initial solutions for that. Thus, we believe that human evaluation is still necessary, similar to text generation tasks. Hence, we perform human evaluation on 50 samples by two expert annotators (of the two T5 models' generated graph outputs), where all 50 samples had a correctly predicted stance and a structurally correct predicted graph (thus results on these samples are not directly comparable to other results in the paper because they are across all samples and not just

[8] We are also collecting multiple ground truth explanation graphs for our evaluation set, which will enable us to check for human correlation to the plausibility metrics.

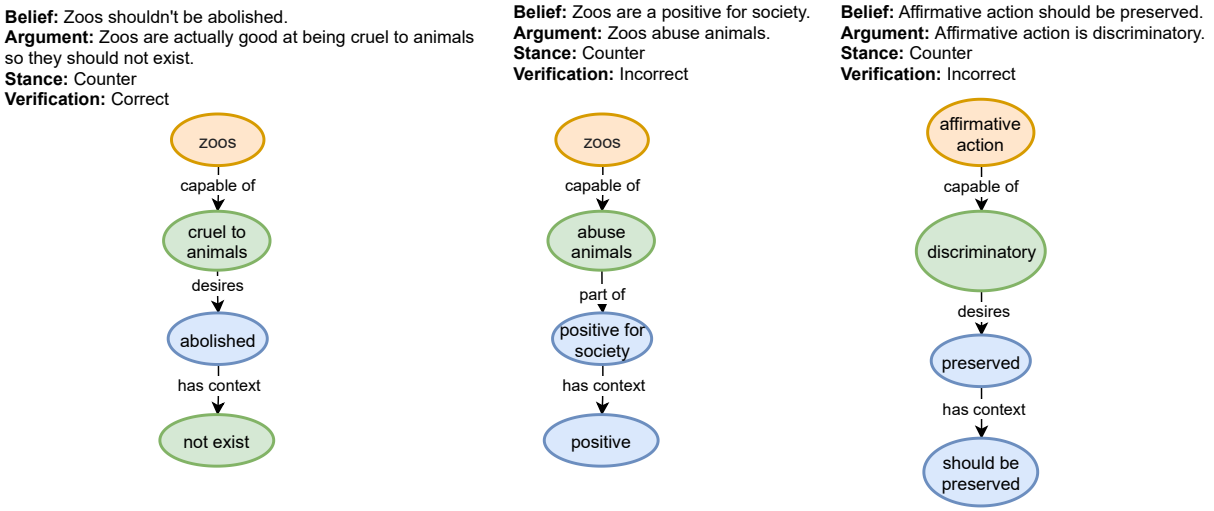


Figure 6: Examples of predicted graph from the T5 reasoning model. The verification term stands for the outcome of human verification while stance is the actual gold stance for this pair.

structurally correct graphs). Results show that out of these structurally correct graphs, humans mark only 36% of the graphs semantically correct for Reasoning-T5 and 34% of the graphs correct for Rationalizing-T5.

8.6 Quantitative Analysis of Generated Explanation Graphs

In order to gain a better understanding of the explanation graphs generated by our models, we compute graph statistics on the structurally correct graphs for the Reasoning-T5 model. Table 9 shows that the predicted graphs contain less number of nodes/edges (4.3/3.3) compared to our gold graphs (5.5/4.6). The average number of external nodes per graph also drops significantly from 1.6 (gold) to 0.3 (pred) which indicates that the model fails to generate novel commonsense concepts to connect the belief and the argument. Additionally, the generated graphs are mostly linear in structure (90%), indicating that pre-trained language models fail to learn and generate complex dependent structures. Finally, only a small percentage of the generated graphs contain external commonsense concepts, broadly pointing to the lack of background knowledge in such graphs.

8.7 Qualitative Analysis of Generated Explanation Graphs

In Figure 6, we qualitatively analyze three randomly chosen examples and the graphs generated by the Reasoning-T5 model. All of them are linear chains with no external commonsense concept. For the first example, given the belief and the explanation graph, human verifiers marked it as a counter graph which matches the gold label and hence the graph is correct. The other two graphs fail in the human verification stage and were marked incorrect. The second graph contains the fact "(abuse animals; part of; positive for society)", where instead of generating a negative relation ("not part of") for connecting the concepts, it chooses the positive counterpart, hence making the graph incorrect. The third graph, similarly, contains the fact "(discriminatory; desires; preserved)" which according to general commonsense beliefs is incorrect. These examples are indicative of the lack of basic commonsense knowledge in these models and we encourage the community to work on developing better models for our task of commonsense explanation graph generation, possibly by adopting a more structured approach and infusing the models with external commonsense knowledge.

9 Conclusion

We proposed EXPLAGRAPHS, a new *generative* and *structured* commonsense-reasoning task on explanation graph generation for stance prediction. For this new task, we also released a benchmarking dataset that was collected using a novel *Collect-Judge-And-Refine* graph collection framework. The collected graphs serve as structured, non-trivial, complete and unambiguous explanations for the task. Our data collection framework is generic and can be potentially used to collect high-quality graph-based data for other NLP tasks. Additionally, we proposed automatic evaluation metrics for

the EXPLAGRAPHS task, and demonstrated the difficulty of generating commonsense-augmented graphical explanations for the stance prediction task through some initial baseline models. We hope our task and dataset will encourage future work on better graph-based commonsense explanation generation.

Acknowledgements

We thank Peter Hase for his helpful feedback. This work was supported by DARPA MCS Grant N66001-19-2-4031, NSF-CAREER Award 1846185, Microsoft Investigator Fellowship, Munroe & Rebecca Cobey Fellowship, and an NSF Graduate Research Fellowship. The views in this article are those of the authors and not the funding agency.

References

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Wen-tau Yih, and Yejin Choi. 2019. Abductive commonsense reasoning. In *International Conference on Learning Representations*.

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.

Michael Boratko, Xiang Li, Tim O'Gorman, Rajarshi Das, Dan Le, and Andrew McCallum. 2020. Protoqa: A question answering dataset for prototypical common-sense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1122–1136.

Faeze Brahman, Vered Shwartz, Rachel Rudinger, and Yejin Choi. 2020. Learning to rationalize for nonmonotonic reasoning with distant supervision. *arXiv preprint arXiv:2012.08012*.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In *NeurIPS*.

Ernest Davis and Gary Marcus. 2015. Commonsense reasoning and commonsense knowledge in artificial intelligence. *Communications of the ACM*, 58(9):92–103.

Leon Derczynski, Kalina Bontcheva, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Arkaitz Zubiaga. 2017. Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 69–76.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. 2020. Eraser: A benchmark to evaluate rationalized nlp models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458.

Maxwell Forbes, Jena D Hwang, Vered Shwartz, Maarten Sap, and Yejin Choi. 2020. Social chemistry 101: Learning to reason about social and moral norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 653–670.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did Aristotle Use a Laptop? A Question Answering Benchmark with Implicit Reasoning Strategies. *Transactions of the Association for Computational Linguistics (TACL)*.

Martin Gleize, Eyal Shnarch, Leshem Choshen, Lena Dankin, Guy Moshkowich, Ranit Aharonov, and Noam Slonim. 2019. Are you convinced? choosing the more convincing evidence with a siamese network. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 967–976.

Shai Gretz, Roni Friedman, Edo Cohen-Karlik, Assaf Toledo, Dan Lahav, Ranit Aharonov, and Noam Slonim. 2019. A large-scale dataset for argument quality ranking: Construction and analysis. *arXiv preprint arXiv:1911.11408*.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. In *2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2018*, pages 107–112. Association for Computational Linguistics (ACL).

Ivan Habernal and Iryna Gurevych. 2016. Which argument is more convincing? analyzing and predicting convincingness of web arguments using bidirectional lstm. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1589–1599.

Momchil Hardalov, Arnav Arora, Preslav Nakov, and Isabelle Augenstein. 2021. A survey on stance detection for mis-and disinformation identification. *arXiv preprint arXiv:2103.00242*.

Kazi Saidul Hasan and Vincent Ng. 2014. Why are you taking this stance? identifying and classifying reasons in ideological debates. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 751–762.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos qa: Machine reading comprehension with contextual commonsense reasoning. 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 2391–2401.

Jena D Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. 2020. Comet-atomic 2020: On symbolic and neural commonsense knowledge graphs. *arXiv preprint arXiv:2010.05953*.

Naoya Inoue, Pontus Stenetorp, and Kentaro Inui. 2020. R4c: A benchmark for evaluating rc systems to get the right answer for the right reason. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6740–6750.

Peter Jansen and Dmitry Ustalov. 2019. Textgraphs 2019 shared task on multi-hop inference for explanation regeneration. In *Proceedings of the Thirteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-13)*, pages 63–77.

Peter A Jansen, Elizabeth Wainwright, Steven Marmorstein, and Clayton T Morrison. 2018. Worldtree: A corpus of explanation graphs for elementary science questions supporting multi-hop inference. *arXiv preprint arXiv:1802.03052*.

Harsh Jhamtani and Peter Clark. 2020. Learning to explain: Datasets and models for identifying valid reasoning chains in multihop question-answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 137–150.

Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. 2020. Qasc: A dataset for question answering via sentence composition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8082–8090.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Xiang Li, Aynaz Taheri, Lifu Tu, and Kevin Gimpel. 2016. Commonsense knowledge base completion. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1445–1455.

Bill Yuchen Lin, Frank F Xu, Kenny Zhu, and Seungwon Hwang. 2018. Mining cross-cultural differences and similarities in social media. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 709–719.

Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. 2020. Commongen: A constrained text generation challenge for generative commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1823–1840.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Hugo Liu and Push Singh. 2004. Conceptnet—a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4):211–226.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Nicholas Lourie, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Unicorn on rainbow: A universal commonsense reasoning model on a new multitask benchmark. *arXiv preprint arXiv:2103.13009*.

Ana Marasović, Chandra Bhagavatula, Jae sung Park, Ronan Le Bras, Noah A Smith, and Yejin Choi. 2020. Natural language rationales with full-stack visual reasoning: From pixels to semantic frames to commonsense graphs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 2810–2829.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41.

Nasrin Mostafazadeh, Aditya Kalyanpur, Lori Moon, David Buchanan, Lauren Berkowitz, Or Biran, and Jennifer Chu-Carroll. 2020. Glucose: Generalized

and contextualized story explanations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4569–4586.

Sharan Narang, Colin Raffel, Katherine Lee, Adam Roberts, Noah Fiedel, and Karishma Malkan. 2020. Wt5?! training text-to-text models to explain their predictions. *arXiv preprint arXiv:2004.14546*.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2019. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Danish Pruthi, Bhuwan Dhingra, Livio Baldini Soares, Michael Collins, Zachary C. Lipton, Graham Neubig, and William W. Cohen. 2020. Evaluating explanations: How much do explanations from the teacher aid students?

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy. Association for Computational Linguistics.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Winogrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8732–8740.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. 2019a. Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3027–3035.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. 2019b. Socialiqa: Commonsense reasoning about social interactions. In *Conference on Empirical Methods in Natural Language Processing*.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. In *Proceedings of ACL*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158.

Cunxiang Wang, Shuailong Liang, Yue Zhang, Xiaonan Li, and Tian Gao. 2019. Does it make sense? and why? a pilot study for sense making and explanation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4020–4026.

Sarah Wiegreffe and Ana Marasović. 2021. Teach me to explain: A review of datasets for explainable nlp. *arXiv preprint arXiv:2102.12060*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Zhengnan Xie, Sebastian Thiem, Jaycie Martin, Elizabeth Wainwright, Steven Marmorstein, and Peter Jansen. 2020. Worldtree v2: A corpus of science-domain structured explanations and inference patterns supporting multi-hop inference. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5456–5473.

Frank F Xu, Bill Yuchen Lin, and Kenny Zhu. 2018. Automatic extraction of commonsense locatednear knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 96–101.

Qinyuan Ye, Xiao Huang, Elizabeth Boschee, and Xiang Ren. 2020. Teaching machine comprehension with compositional explanations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1599–1615.

Mo Yu, Shiyu Chang, Yang Zhang, and Tommi Jaakkola. 2019. Rethinking cooperative rationalization: Introspective extraction and complement control. 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 4085–4094.

Omar Zaidan, Jason Eisner, and Christine Piatko. 2007. Using "annotator rationales" to improve machine learning for text categorization. In *Human language technologies 2007: The conference of the North American chapter of the association for computational linguistics; proceedings of the main conference*, pages 260–267.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.

Hongming Zhang, Xinran Zhao, and Yangqiu Song. 2020. Winowhy: A deep diagnosis of essential commonsense knowledge for answering winograd schema challenge. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5736–5745.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Appendix

A.1 Experimental Setup

We train all models using the Hugging Face transformers library (Wolf et al., 2019).[9] Across all our RoBERTa, BART and T5 models, we use a batch size of 8. For RoBERTa, we use an initial learning rate of $10^{-}5$, while for BART and T5, we use learning rates of $3 * 10^{-}5$. For decoding the graphs from the generative models, we use standard beam search decoding with beam size of 4. We train all models for a maximum of 10 epochs and the best model is chosen based on our dev set. We use a random seed of 42 across all our experiments. All experiments are performed on one V100 Volta GPU.

A.2 Topics and Commonsense Relations Used in ExplaGraphs

In Figure 7, we show the full list of debate topics used in our data collection process. The train split consists of 53 topics, while the dev and the test splits contain 9 topics each. Figure 8 shows all the commonsense relations used for our explanation graph creation. We broadly choose the relation set from ConceptNet (Liu and Singh, 2004), while removing generic relations like "related to" and adding a negative counterpart for every positive relation to enable the composition of supportive and counter graphs.

[9] https://github.com/huggingface/transformers

A.3 Stage 1 Task Instructions and Interface

In Fig. 9, we show the instructions for our pre-HAMLET part of Stage 1 of data collection framework. Fig. 10 shows the instructions for the HAMLET rounds. Finally, in Fig. 11, we show the interface for our stance label verification, given the belief and the argument.

A.4 Stage 2 Task Instructions and Interface

Task	Pay/HIT (in cents)
Pre-HAMLET Collection	25
HAMLET Collection	25
Stance Verification	5
Graph Collection	45
Graph Refinement	45
Graph Verification	10

Table 10: Payment per HIT (in cents) for each of our tasks on MTurk.

Fig. 12 shows the detailed instructions that we provide to the annotators for commonsense explanation graph creation. We start by explaining the overall motivation and goal of this task, followed by the definitions of commonsense fact, concept, and relation. As part of the guidelines, we provide the detailed steps to perform this task and the list of structural constraints on the explanation graphs. We also remind the workers to verify their own graphs before submitting by following three basic steps of stance inference from the graphs. We also provide examples of disconnected and cyclic graphs to help them understand structurally incorrect graphs. In Fig. 13, we show the instructions provided for verifying the semantic correctness of our commonsense explanation graphs. In this stage, we refer to explanation graphs as argument graphs since our graphs are extended structured arguments. We provide annotators will only the belief and the argument graph, and ask them to choose between incorrect, support and counter labels. We also provide examples of semantically incorrect graphs. Fig. 14 shows the corresponding interface for graph verification. Finally, in Fig. 15, we show the instructions of graph refinement where we also provide some broad guidelines of how to refine the graphs. In Table 10, we show the payment per HIT for each of our tasks, broadly maintaining an hourly pay of 12-15$.

Train Topics

We should cancel pride parades
We should ban algorithmic trading
We should subsidize stay-at-home dads
We should introduce compulsory voting
We should subsidize space exploration
We should abolish the right to keep and bear arms
We should abolish the Olympic Games
Holocaust denial should be a criminal offence
We should abolish the three-strikes laws
We should prohibit school prayer
We should adopt gender-neutral language
We should ban cosmetic surgery for minors
We should abandon the use of school uniform
We should adopt a multi-party system
Payday loans should be banned
Blockade of the Gaza Strip should be ended
We should legalize cannabis
Homeopathy brings more harm than good
We should ban targeted killing
Assisted suicide should be a criminal offence
The use of public defenders should be mandatory
We should abandon television
We should prohibit flag burning
We should limit judicial activism
We should end the use of economic sanctions
We should end mandatory retirement

Train Topics

We should ban missionary work
We should ban the Church of Scientology
We should ban naturopathy
We should subsidize journalism
The vow of celibacy should be abandoned
We should adopt a zero-tolerance policy in schools
Surrogacy should be banned
We should ban telemarketing
We should legalize sex selection
We should legalize prostitution
We should ban whaling
We should ban private military companies
We should legalize organ trade
We should subsidize student loans
We should prohibit women in combat
We should end racial profiling
We should abolish intellectual property rights
We should ban factory farming
Intelligence tests bring more harm than good
We should ban the use of child actors
We should abolish safe spaces
We should subsidize Wikipedia
Foster care brings more harm than good
Social media brings more harm than good
We should abolish capital punishment
We should limit executive compensation

Dev Topics

We should stop the development of autonomous cars
Homeschooling should be banned
We should subsidize vocational education
We should abolish zoos
We should oppose collectivism
We should fight for the abolition of nuclear weapons
We should ban fast food
We should end affirmative action
We should legalize polygamy

Test Topics

We should abandon marriage
We should ban cosmetic surgery
We should adopt an austerity regime
We should fight urbanization
We should subsidize embryonic stem cell research
Entrapment should be legalized
We should ban human cloning
We should close Guantanamo Bay detention camp
We should adopt atheism

Figure 7: The complete list of debate topics used in our data collection process.

antonym of	has subevent
synonym of	not has subevent
at location	part of
not at location	not part of
capable of	has context
not capable of	not has context
causes	has property
not causes	not has property
created by	made of
not created by	not made of
is a	receives action
is not a	not receives action
desires	used for
not desires	not used for

Figure 8: The complete list of relations used for explanation graphs.

A.5 Examples from EXPLAGRAPHS

In this section, we show some of the examples from the EXPLAGRAPHS. Please see Figure 16, 17, 18, 19, 20, 21, 22, 23, 24.

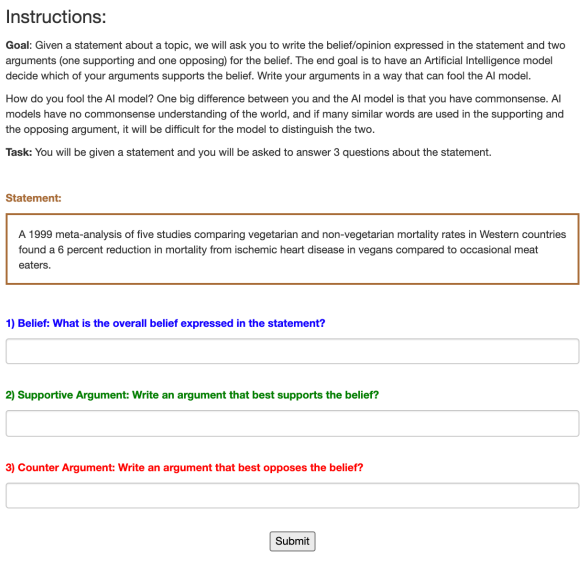


Figure 9: Interface showing the instructions for collecting pre-HAMLET belief and argument (support and counter) pairs on MTurk, given a prompt about one of the debate topics.

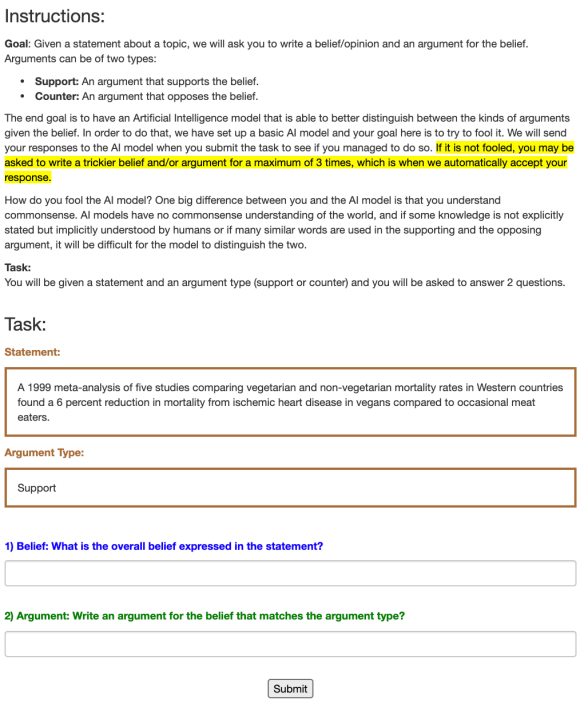


Figure 10: Interface showing the instructions for collecting HAMLET belief and argument pairs on MTurk, given a prompt about one of the debate topics and the target stance label (support or counter).

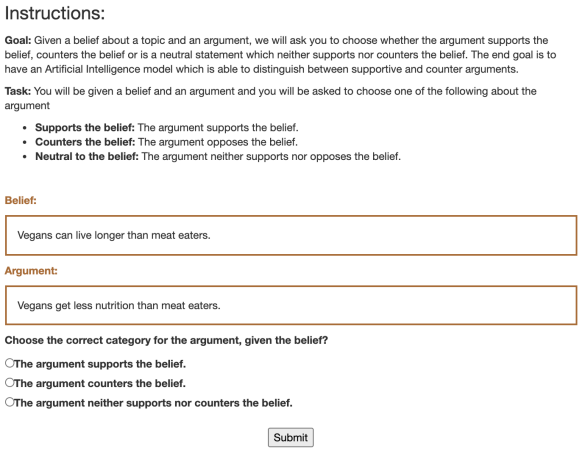


Figure 11: Interface showing the instructions for verifying belief and argument pairs on MTurk. We keep only those pairs which have majority stance label support or counter across five verifiers.

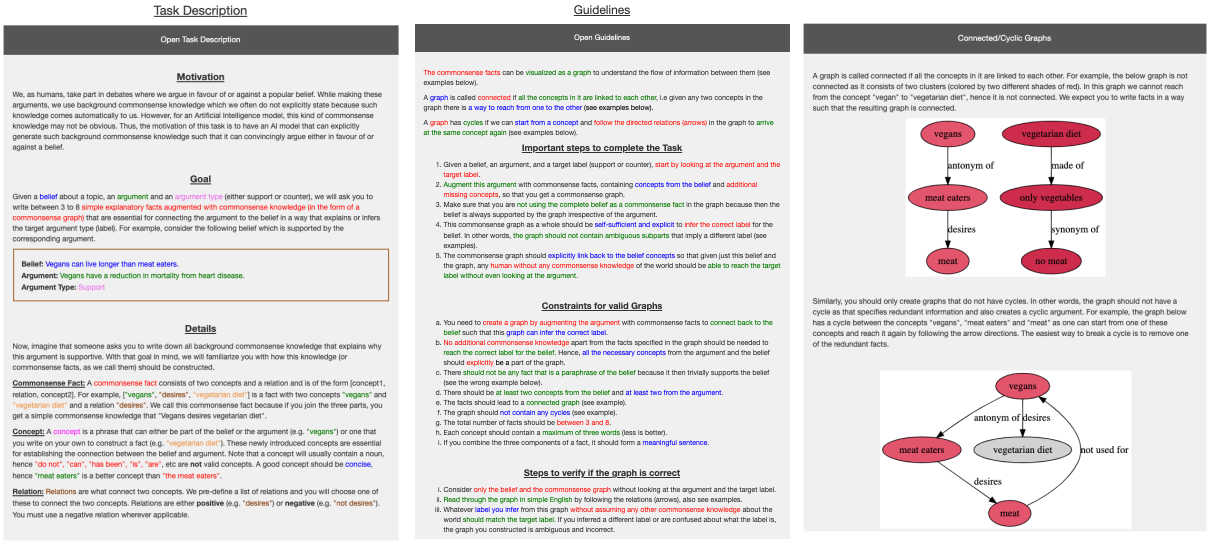


Figure 12: Instructions for commonsense explanation graph creation: We start by explaining the overall motivation and goal of this task, followed by the definitions of commonsense fact, concept, and relation. As part of the guidelines, we provide the detailed steps to perform this task and the list of structural constraints on the explanation graphs. We also remind the workers to verify their own graphs before submitting by following three basic steps of stance inference from the graphs. Since workers are required to fix their graphs if they are not connected DAGs, we also provide examples of disconnected and cyclic graphs.

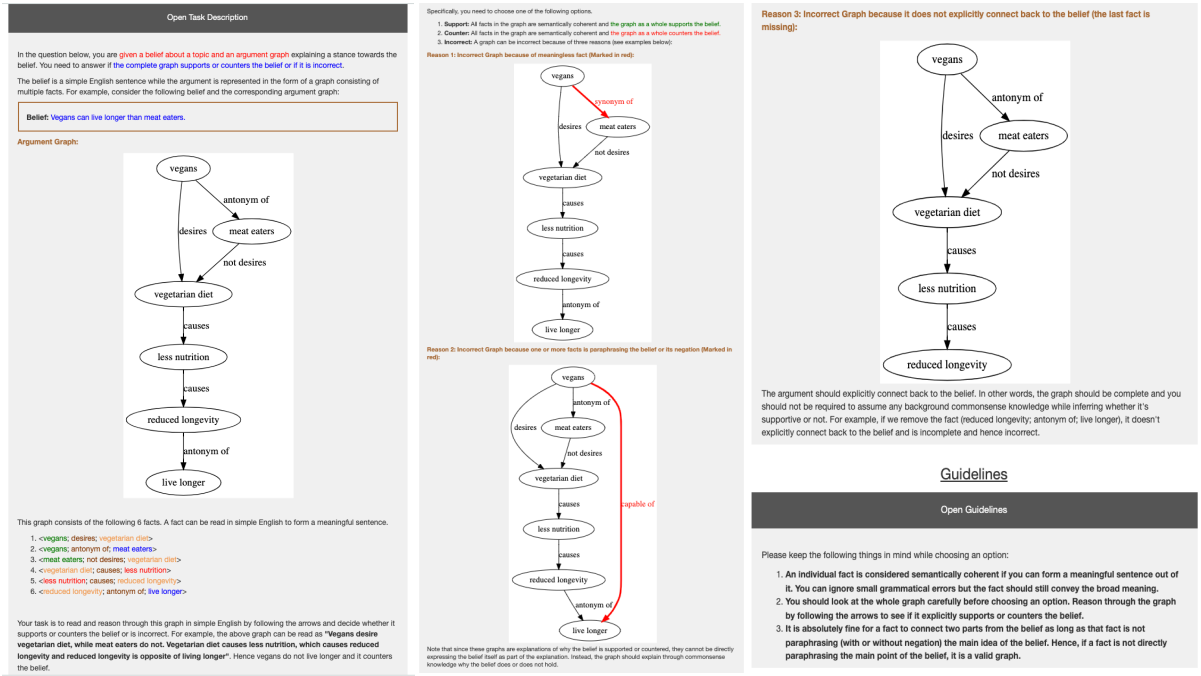


Figure 13: Instructions for commonsense graph verification: Explanation graphs are treated as augmented structured arguments for this task and hence referred to as argument graphs. Given a belief and the argument graph, workers are required to choose between incorrect, support and counter labels. We begin by visually explaining what an argument graph is, and also show examples of incorrect graphs. To ensure good inter-annotator agreement and that the semantically incorrect graphs are identified correctly, we also provide some general guidelines for performing this task.

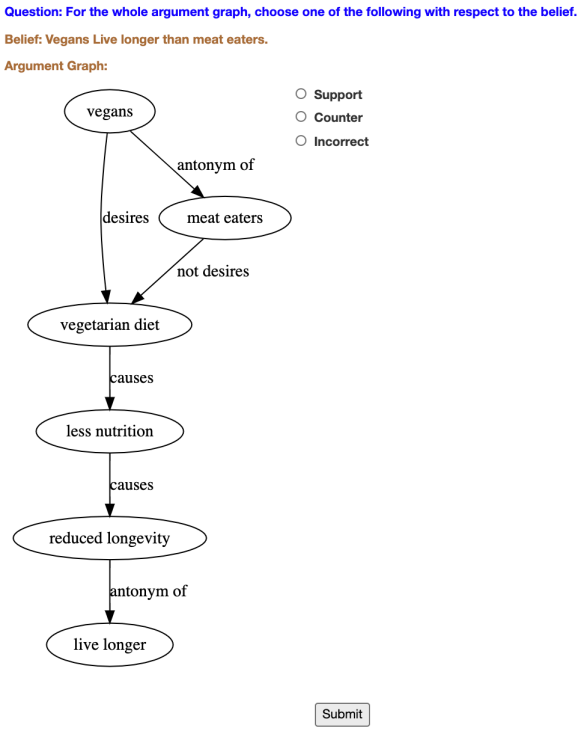


Figure 14: Interface for Stage 2 graph verification

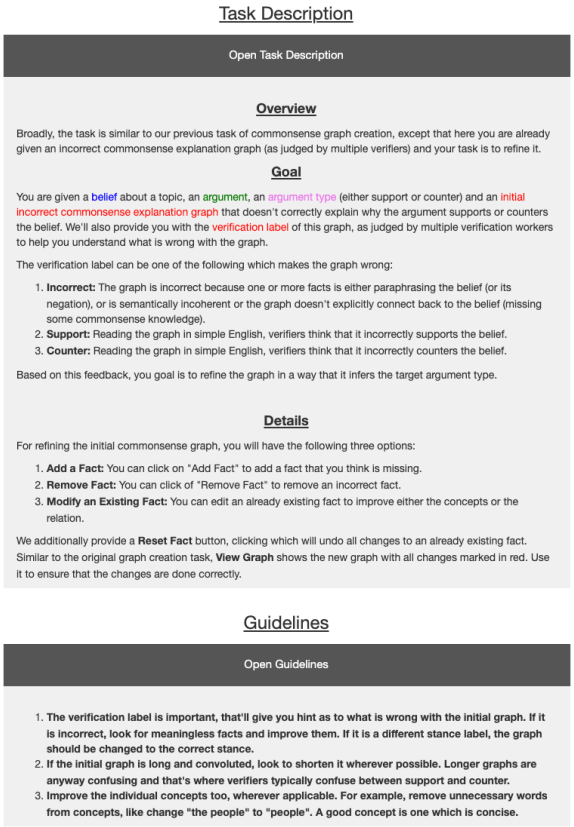


Figure 15: Instructions for Stage 2 graph refinement.

Belief: Celibacy should be respected as an expression of belief.
Argument: Vows of celibacy are often related to religious beliefs.
Stance: Support

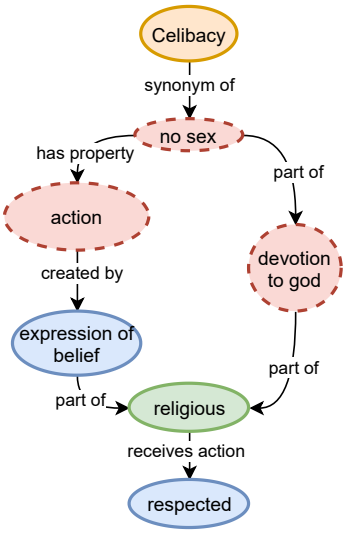


Figure 16: Example-3

Belief: Entrapment should be legal.
Argument: Entrapment catches terrible people.
Stance: Support

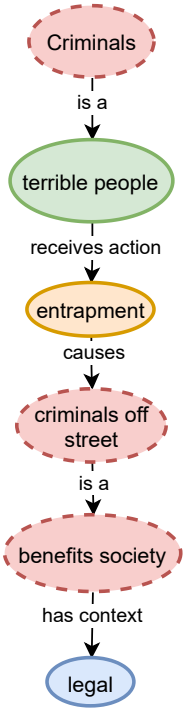


Figure 17: Example-4

Belief: Organ transplant is important.
Argument: A patient with failed kidneys might not die if he gets organ donation.
Stance: Support

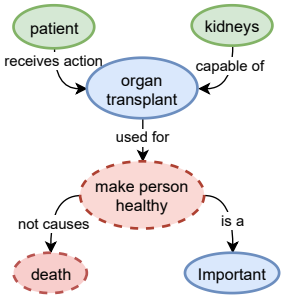


Figure 18: Example-5

Belief: Autonomous car development should end.
Argument: Autonomous cars would be better than humans.
Stance: Counter

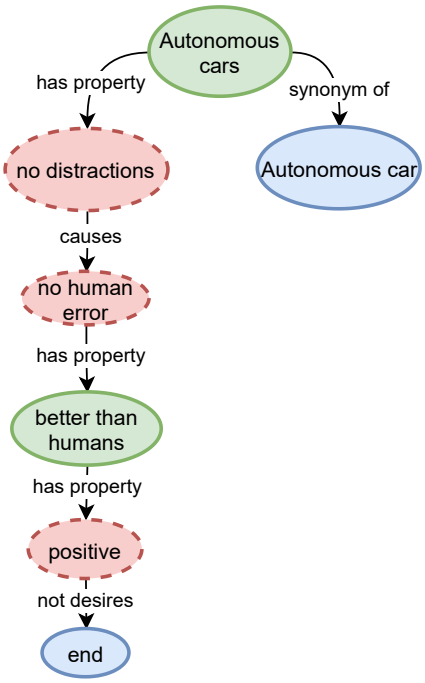


Figure 19: Example-6

Belief: Allowing organ trade does harm to the poor.
Argument: If we allow organ trade, the poor can more easily pay to acquire needed resources.
Stance: Counter

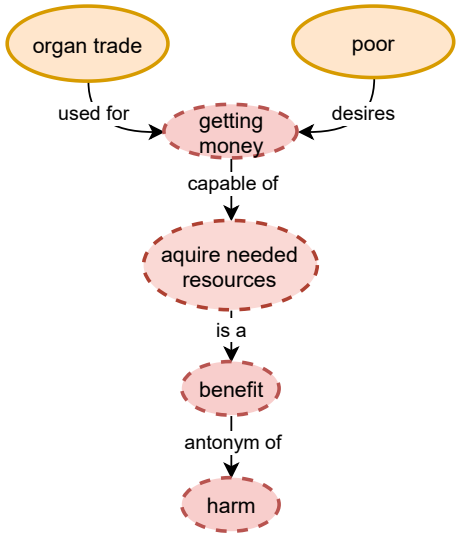


Figure 20: Example-7

Belief: Marriage is extremely important for strong families.
Argument: Marriage has been a staple in society for centuries.
Stance: Support

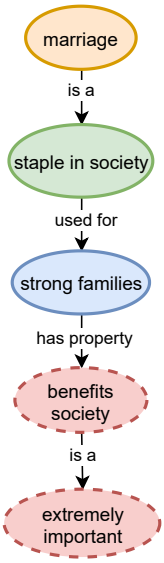


Figure 21: Example-8

Belief: Cosmetic surgery should not have an age requirement.
Argument: Young people with traumatic accidents may need reconstructive surgery just as much as an adult would.
Stance: Support

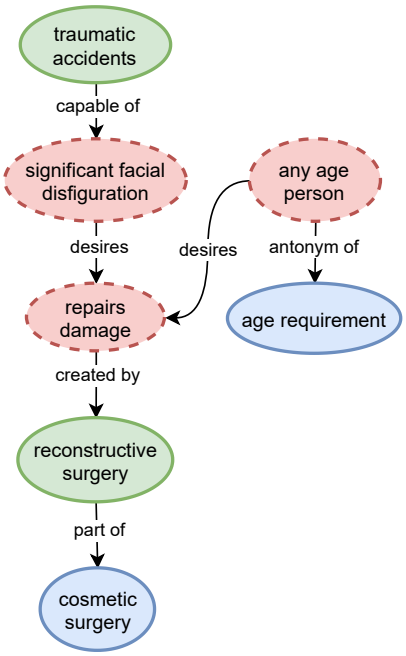


Figure 22: Example-9

Belief: Plastic surgery should not be shamed.
Argument: Plastic surgery is harmful to one's self esteem.
Stance: Counter

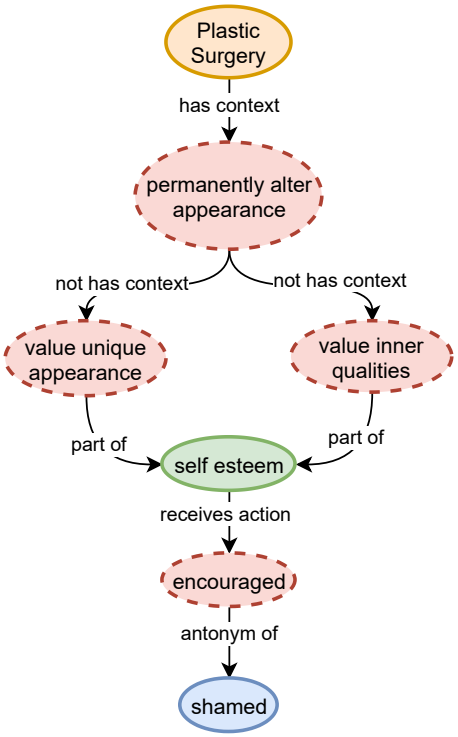


Figure 23: Example-10

Belief: Marriage does not mean much.
Argument: Marriage is the backbone of society.
Stance: Counter

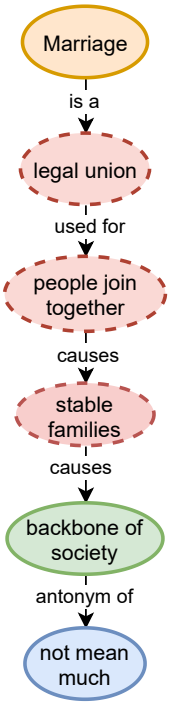


Figure 24: Example-11