Enhancing Systematic Decompositional Natural Language Inference Using Informal Logic

Nathaniel Weir[♠*] **Kate Sanders**[♠] **Orion Weller**[♠] **Shreya Sharma**[♠]
Dongwei Jiang[♠] **Zhengping Jiang**[♠] **Bhavana Dalvi Mishra**[†] **Oyvind Tafjord**[†]
Peter Jansen[‡†] **Peter Clark**[†] **Benjamin Van Durme**[♠]
[♠]Johns Hopkins University [†]Allen Institute for AI [‡]University of Arizona
{nweir, ksande25, vandurme}@jhu.edu

Abstract

Recent language models enable new opportunities for structured reasoning with text, such as the construction of intuitive, proof-like textual entailment trees without relying on brittle formal logic (Tafjord et al., 2022; Weir et al., 2024). However, progress in this direction has been hampered by a long-standing lack of a clear protocol for determining what *valid compositional entailment* is. This absence causes noisy datasets and limited performance gains by modern neuro-symbolic engines. To address these problems, we formulate a consistent and theoretically grounded approach to annotating decompositional entailment and evaluate its impact on LLM-based textual inference. We find that our new dataset, RDTE (Recognizing Decompositional Textual Entailment), has a substantially higher internal consistency (+9%) than prior decompositional entailment datasets. We also find that training an RDTE-oriented entailment classifier via knowledge distillation and employing it in an entailment tree reasoning engine significantly improves both accuracy and proof quality, illustrating the practical benefit of this advance for textual inference.

1 Introduction

Textual inference using entailment trees has emerged as a promising avenue for explainable AI in domains requiring complex reasoning (Dalvi et al., 2021; Bostrom et al., 2022). Inspired by classical symbolic reasoners that produce proof trees of a hypothesis, recent entailment tree systems (e.g. NELLIE (Weir et al., 2024) or IRGR (Ribeiro et al., 2022)) explain a hypothesis by constructing, via an LM-based generative search algorithm, a tree of recursive natural language (NL) decompositions whose children entail their parents. Entailment trees provide a natural and granular medium for explaining complex multi-step decisions to users and create opportunities to ground reasoning to large NL knowledge corpora (Weir et al., 2024).

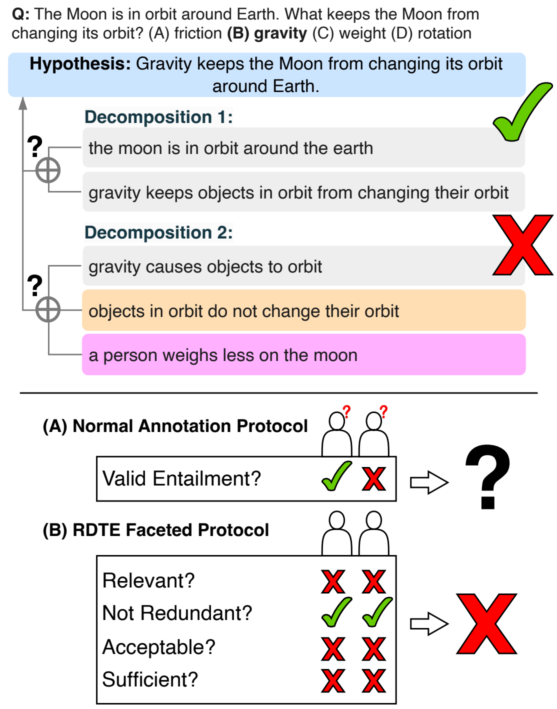


Figure 1: **(Upper)** Two hypothesis decompositions suggested by an LLM. The first makes an argument that is generally acceptable to a human. The second contains a fact that is not always true and another that is irrelevant to the entailment. Recognizing such an invalid decomposition is core to recent neuro-symbolic reasoning algorithms, but LLMs struggle at the task. **(Lower)** Ambiguous definitions of entailment have hampered progress in annotating data to improve the models. We find that a faceted definition yields both a clean dataset (RDTE) and significant downstream task improvements.

Entailment tree reasoners are most useful to humans if they are *right for the right reasons*: They not only give correct answers, but each tree step is a valid deduction supporting its hypothesis. While some tree generators leverage classifiers for recognizing textual entailment (RTE; Dagan et al., 2005) to identify which hypothesis decompositions are illogical and should be discarded (Yang et al., 2022),

[*] Work done in part as intern for Allen Institute for AI.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 9458–9482
November 12-16, 2024 ©2024 Association for Computational Linguistics

it is still common for systems to arrive at the correct conclusion while presenting a flawed decomposition, or to accept incorrect conclusions based on faulty logic. Thus, the task of *recognizing decompositional textual entailment* is a major barrier to progress on tree-based reasoners.

Work on entailment trees has largely avoided this critical question of reasoning validity. Existing evaluations focus on tree reconstruction (Ribeiro et al., 2023) and QA performance, avoiding considerations of whether the quality of model-generated trees aligns with the accuracy of the answers. This is largely because reasoning quality is difficult to assess. Evidence suggests humans think of deductive validity less stringently than formal symbolic axioms; they accept explanations that are more or less incomplete (Sulik et al., 2021; Tan, 2022). Attempts to annotate datasets (Tafjord et al., 2022; Clark et al., 2023) consequently suffer from inconsistent labels, as opinions differ between annotators over what constitutes validity. Without careful instruction, it is hard for annotators to determine where to set their threshold for acceptability on the spectrum from "strict deductive inferences" to "fallible common-sense inferences" (Gubelmann et al., 2023). Thus, asking for a single binary decompositional entailment judgment yields a difficult and noisy annotation task (Figure 1, (A)). This leads us to pursue a more consistent protocol for measuring the goodness of one decomposition over another.

The existing discourse surrounding NLI (e.g., Manning (2006)) has not addressed the sort of complex decompositional textual entailments encountered in a recursive tree search algorithm for current QA datasets. Towards addressing this gap, we propose an evaluation centered on the notion that **a valid decomposition is a valid argument for why the hypothesis should be believed**. We take inspiration from the "Relevance, Acceptability, and Sufficiency" criteria for a logically good argument developed in the field of informal logic (Johnson and Blair, 1977; Groarke, 2022), and design an approach to assessing entailment in a principled manner. We annotate items on ordinal scales targeting different facets of argument analysis, yielding a well-defined and consistent process. We use this protocol to collect over 1000 expert annotations of decompositions generated during entailment tree searches. Through experiments, we find that models trained on previous compositional entailment datasets and LLMs like GPT fall short of human performance on the dataset, which we term RDTE (Recognizing Decompositional Textual Entailment).

We also find that our annotation protocol can serve as a powerful LM prompting mechanism: Given the same instructions as our human annotators, GPT-4 (OpenAI, 2023) can serve as a "teacher" in a knowledge distillation pipeline that annotates reasoning traces extracted from a tree search algorithm. For this purpose, we released a large artifact (24K items per domain) of GPT-4's annotations. We illustrate the effectiveness of student models trained via this pipeline as the linchpin of TREEWISE, a novel entailment tree engine that performs hypothesis proving via decompositional entailment search. Inspired by the recent NELLIE engine (Weir et al., 2024), TREEWISE leverages in-context learning, forward chaining, branch consolidation, and most importantly, improved decompositional entailment recognition. It surpasses tree-producing approaches on established benchmarks like EntailmentBankQA but also adapts to complex tasks such as HotpotQA that require reasoning over less structured knowledge sources like Wikipedia. Using the knowledge-distilled models as reasoning verifiers improves TREEWISE's QA accuracy *and* raises the quality of the trees it produces. Our contributions are thus:

- **A new entailment challenge set**, RDTE, crafted via an informal logic-inspired protocol, tasking models to validate hypothesis decompositions
- **A knowledge distillation pipeline** that teaches student models to discriminate **what** and **why** decompositions are invalid in a given domain.
- **A new entailment tree-generating inference engine**, TREEWISE, which outperforms existing tree-based QA methods while generating higher-quality trees in the process. This engine shows to benefit from RDTE knowledge distillation and illustrates the viability of entailment-based reasoning using current LLM technologies.[1]

2 Related Work

Neural Entailment Engines Recent papers have explored LM-based algorithms for generating entailment trees given a set of support facts (Dalvi et al., 2021; Ribeiro et al., 2022; Yang et al., 2022; Bostrom et al., 2022; Hong et al., 2022). There is

[1]Data, code, and models can be found at https://github.com/JHU-CLSP/treewise.

also growing literature on backward- and forward-chaining entailment tree algorithms for performing QA (Tafjord et al., 2022; Weir et al., 2024; Hong et al., 2023; Sanders et al., 2024). These approaches have often grappled with the challenge that current popular LMs frequently generate invalid entailment steps and fail to recognize these errors. Previous generations have also explored symbolic entailment search algorithms, e.g., Nutcracker (Bos, 2014) and NaturalLI (Angeli and Manning, 2014), though these formal logic-based theorem-proving approaches generally failed to scale to the complex inferences necessary to verify the entailments necessary for current QA datasets (Clark et al., 2018).

Annotating Textual Entailment Datasets The PASCAL RTE Challenge (Dagan et al., 2005), SNLI (Bowman et al., 2015) and MNLI (Williams et al., 2018) have led to numerous NLI leaderboards. These datasets (A) are generally not tied to downstream reasoning tasks; (B) commonly target uncontroversial items for which annotators have high agreement without needing granular instructions. The BaRDa dataset (Clark et al., 2023) throws out a large fraction of decomposition annotations because of low agreement. In contrast, RDTE specifically targets the hard-to-evaluate items that unavoidably manifest during systematic reasoning algorithms. Our use of ordinal annotation draws from JOCI (Zhang et al., 2017), a collection of common sense inferences annotated on a 5-point scale of likeliness. Previous attempts at recognizing compositional entailment have used QASC (Khot et al., 2020) and eQASC (Jhamtani and Clark, 2020), two large-scale, noisy sets of sentence pair compositions.

Annotator Disagreement Various works have faced the challenge of annotator disagreement for reasoning tasks. AmbiFC (Glockner et al., 2021), concerned with whether a piece of evidence supports a given claim, explores the common factors causing disagreements. Pavlick and Kwiatkowski (2019) found persistent patterns of disagreement amongst RTE judgments not simply due to noise; efforts such as the UNLI protocol (Chen et al., 2020) aim to address this concern by modeling annotations on a logistic probability scale.

Computational Argumentation Existing work has explored methods for computational argumentation; one subfield is argumentation mining (Palau and Moens, 2009), which aims to detect and analyze the arguments in a passage. Jin et al. (2022) collect a dataset of items exhibiting 14 types of fallacies. Stab and Gurevych (2017) hand-annotate the sufficiency of argumentative essays using RAS criteria but are not geared towards entailment.

Assessing Explanations Our work builds on literature evaluating the role and quality of model-generated explanations (Wiegreffe and Marasovic, 2021; Tan, 2022). We add to this discourse by investigating how to improve the quality of decompositional entailment items that drive reasoning for complex tasks. Similar to ours is the EEV methodology (Valentino et al., 2021), under which an explanation is translated into logical form and then assessed by a trained semanticist. RDTE requires neither the translation nor the trained semanticist.

3 Decompositional RTE

3.1 Task Definition

An entailment tree comprises one or multiple hypothesis decompositions. Each constitutes one instance of decompositional RTE, which we define as follows: Given the question Q, hypothesis h, and premises $[p_1, \ldots, p_n]$, would proving the premises be contextually sufficient to prove h?

As described below, we collect different ordinal scores for each item using criteria from informal logic. Each facet informs the ultimate sufficiency label, for which we evaluate models in §5.

3.2 Background: RAS Criteria

A seminal framework developed by informal logicians to replace strict deductive logic criteria is known as RAS: **Relevance**, **Acceptability**, and **Sufficiency** (Johnson and Blair, 1977). Much like RTE work, RAS assesses whether a hypothesis can be accepted by a specific audience[2] based on a list of premises. Noting that each element is subject to substantial academic debate (different systems of informal logic define them differently), we review each criterion as we interpret them for this work:

Relevance of premise A concerning conclusion B is defined as whether the truth of A makes a difference to the truth of B. (Blair, 2012). The

[2]In this work, we pose the "audience" of the entailment argument following Manning (2006): "[P]eople that are awake, careful, moderately intelligent and informed, and with reasonable document interpretation expertise, but not by semanticists or similar academics. These people would use whatever background and real-world knowledge that they usually bring to interpreting texts."

extent of this difference can vary; e.g. whether (A) "the earth has oxygen" is true technically has relevance to (B) birds can fly (since birds breathe oxygen), but is less relevant than (A') birds have wings. Relevance is particularly important to a recursive reasoner, as one would not want to waste time proving an irrelevant statement like (A) when proving (B).

Acceptability of a premise is normatively "worthy of acceptance," which can mean either its ostensible truth value or-–in the absence of universal factuality–that in the relevant context, the arguer and the argument recipient accept it to be true. This introduces a hiccup for recursive reasoning algorithms, for which the factuality of a decomposition's premises is commonly determined via search *after* validating the decomposition itself. We ultimately choose to annotate one subset of RDTE for premise factuality while not doing so for the other.

Sufficiency is "the property of an argument's premises of supplying all the grounds that are needed to make it reasonable to believe its conclusion." (Johnson and Blair, 1977). This is left intentionally vague; Blair (2012) notes "the criterion of sufficiency, for justificatory arguments, is best seen as a placeholder for whatever version and standards of sufficiency are appropriate for the particular situation in question." In this way, we should consider sufficiency to be a question and problem-specific criterion: **the grounds for valid entailment are inherently domain-specific**. Blair (2012) notes the dependent relationship between sufficiency and the other two criteria: sufficient presumes acceptability and relevance.

3.3 Implementing RAS for RTE Annotation

We draw inspiration from these criteria to construct a protocol that investigates a decomposition like the ones shown in Figure 1 for each RAS component in turn. An important aspect of these normative terms is that they are all scalar: a premise can be more or less relevant and acceptable, and the sufficiency of an argument composed of premises could always be strengthened by adding more premises. In a departure from works such as Tafjord et al. (2022) who collect binary factuality and "reasoning correctness" judgments, we collect RAS judgments on an ordinal scale (Zhang et al., 2017) from 1 to 5, where each score is assigned a specific set of conditions. We also collect an ordinal judgment for **redundancy**, which is not strictly a component of RAS assessment; we observe that redundant premises are particularly problematic for entailment tree search, as proving the same information twice wastes search budget. We implement redundancy as "conditional irrelevance":

- If removing a premise *in isolation* doesn't change the extent of the entailment, then it's **irrelevant**.
- If removing a premise *in the presence of the other decomposition premises* does not change the extent of the entailment, then it's **redundant**.[3]

The sufficiency label most directly resembles a typical RTE label, and we evaluate models based on it in later sections. Following Blair (2012)'s observation, the sufficiency judgment is highly dependent on the other criteria; our annotation directions, found in §B, include a substantial list of 30+ conditions around a decomposition that indicate what the sufficiency score should be based on other criteria scores, e.g. "Did you give any fact a 3/5 or lower for redundancy, factuality, or relevance? (Yes = max 4 sufficiency)".

4 Data Collection

We seek to construct a dataset of decompositional entailment judgments that a reasoning system might encounter while performing QA. Following Tafjord et al. (2022), we found it apt to annotate decompositions generated during a model's reasoning search traces over training questions. We use the backward chaining process of the tree search algorithm TreeWise, which is introduced in §6. Each item is a hypothesis and a set of 2-3 premises that the model proposes might conjunctively entail the hypothesis in the context of a given question. To collect a representative sample of hypotheses, we pull from three classes:

1. **Top-level correct hypotheses** representing the right answer options for multiple-choice questions. These are important to annotate so that models are *right for the right reasons.*
2. **Recursive correct hypotheses** LLM-generated to decompose the top-level correct hypotheses
3. **Top-level incorrect hypotheses** representing the incorrect answer deemed closest to correct by GPT-4. These are important to annotate so that models are *not wrong for the wrong reasons.*

We collect a mixture of GPT-4 and ChatGPT-generated decompositions using a suite of different

[3]An irrelevant premise is thus by definition also redundant.

styles of prompts described in §A.2. We generated decompositions for hypotheses[4] in two task domains: multiple choice science QA in **ARC** (Clark et al., 2018) and multi-hop QA over Wikipedia in **HotpotQA** (Yang et al., 2018). While Hotpot questions are constructed over public world knowledge, the esoteric individual facts (e.g. the birth year of some actor) are not common knowledge among humans, and it would be a stretch for the audience of an argument made by a Hotpot decomposition to be expected to know each premise's factuality. We thus do not annotate Hotpot decompositions for factuality, but annotate the other facets as normal.

4.1 Annotation Process

Four of this paper's authors, all highly proficient or native English speakers, annotated 1000 decompositions total with two-way redundancy. One author annotated all items. We first annotated several examples together to develop a list of 30+ condition checks that indicate certain RAS scores. We then annotated the rest of the dataset independently.

While the dataset is annotated for sufficiency on a 5-point scale, we found a threshold of ≥ 4 fitting to evaluate models under binary entailment metrics. This corresponds to items for which the only acceptable flaws are (A) a 3rd redundant premise in an otherwise perfect 2-premise entailment or (B) minor missing information that does not affect lay reasoning. To arrive at a single clean entailment label for evaluation, we reconciled disagreements by discussing all items for which its two annotated sufficiency scores were on either side of 3.5. The vast majority of disagreements were due to human error, e.g., missing a condition or not recognizing a particular flaw in reasoning.

Comparison to Existing Data Before reconciling disagreements, we measured raw annotation agreement and compared it to recent attempts to collect decompositional entailment labels. Tafjord et al. (2022) collected annotations for 3.7K decompositions generated by **Entailer**, while Clark et al. (2023) did so for 24.4K items generated by **GPT3**.[5] The instructions given to annotators for these datasets are very high-level, simply asking whether the "reasoning goes wrong." We believe this creates a much lower threshold for acceptability than the RDTE protocol; these datasets are majority labeled "entailment," but on manual inspection, we found numerous items that exhibited redundant, irrelevant, and fallacious arguments.

Nevertheless, **RDTE has a higher internal annotator consistency rate** than these datasets, with a raw agreement rate 9% higher and Krippendorf α 19 points higher than the previous best.

	RDTE (ours)	Entailer	GPT3
Raw agreement	**.79**	.70	.61
Krippendorf α	**.53**	.34	.31

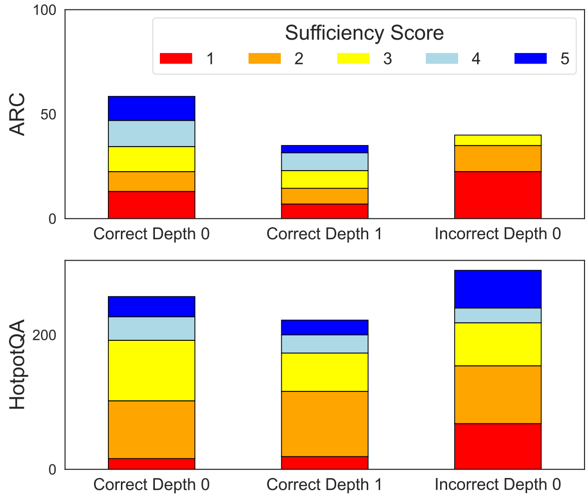


Figure 2: Distribution of the 1000 entailment labels in RDTE. Instead of binary entail/non-entailment, we annotate on a 5-point ordinal scale. To evaluate binary judgment models, we treat $\geq$4 as positively labeled.

4.2 RDTE Analysis

Figure 2 depicts the breakdown of sufficiency labels within the RDTE ARC (267) and Hotpot (775) decompositions. While the ordinal scores are relatively well distributed, only 27% of the dataset is labeled a 4 or 5, creating a large binary label imbalance. This statistic highlights the importance of performing well on this task: 3 out of every 4 decompositions generated by ChatGPT and GPT-4 did not pass our sufficiency rubric.

Figure 3 shows example decomposition annotations from the dataset. 49% of RDTE items had at least one premise rated as irrelevant ($\leq 3/5$); 36% had at least one rated as redundant. 28% of the items with a factuality rating had at least one nonfactual premise. We note that *none of the incorrect ARC hypothesis decompositions were labeled as sufficient*; this highlights that the RDTE pro-

[4] We created hypotheses by declarativizing (Demszky et al., 2018) answer options using GPT-4. For HotpotQA, we first had GPT-4 synthesize incorrect answer options.

[5] The BaRDa dataset comprises a specifically filtered subset of these two datasets that exhibit maximal annotator agreement. We obtained the pre-filtered data from the authors for comparison with our work.

	Fa	Rel	Red
Q: The 2005 film Remedy featured Frank Q: Vincent from The Sopranos and several mob movies by which acclaimed director? (A) Jonathan Demme, (B) Martin Scorsese, (C) Ralph De Vito, (D) Steven Hilliard Stern			
H: The 2005 film Remedy featured Frank Vincent from The Sopranos and several mob movies by the acclaimed director Ralph De Vito.			
P1: Frank Vincent appeared in The 2005 film Remedy.	–	5	5
P2: Frank Vincent appeared in The Sopranos and several mob movies	–	5	5
P3: Ralph De Vito directed several mob movies.	–	5	5
Sufficiency: 3. The facts are relevant and nonredundant, but they do not link the mob movies by De Vito to those that featured Vincent. Substantial missing information is a 3.			
Q: Which process best explains how the Grand Canyon became so wide? (A) folding, (B) erosion, (C) deposition, (D) sedimentation			
H: Erosion best explains how the Grand Canyon became so wide.			
P1: Changes in the Grand Canyon's landscape include becoming wider.	5	5	5
P2: Erosion is one of the processes that can change a landscape.	5	5	5
P3: The Grand Canyon is a landscape.	5	3	5
Sufficiency: 2. The facts are relevant and nonredundant, but they do not establish a direct causal relationship between erosion and the Grand Canyon becoming wider. Removing P3 does not strongly impact the extent of the entailment. Redundant P + missing information is a 2.			
Q: Is Ordos City more west than Yangzhong? (A) No, (B) Same longitude, (C) yes			
H: Ordos City is located in the western part of China.			
P1: Ordos City is predominantly rural.	-	3	5
P2: Predominantly rural areas in China are often found in the western part of the country.	-	3	5
Sufficiency: 2. The argument commits a fallacy of division by assuming that because predominantly rural areas in China are often in the west, Ordos City, being rural, must also be in the western part. This reasoning does not adequately support the conclusion without specific geographic evidence. Fallacious reasoning and irrelevant premises is a 2.			

Figure 3: Example RDTE annotations.

tocol, when including the factuality facet, does a comprehensive job of systematically ruling out decompositions for which there is necessarily some issue (else the hypothesis must be correct). Contrastly, there *are* positively-labeled decompositions of incorrect hypothesis for Hotpot, because we did not annotate for factuality. There are thus decompositions that, were their premises true, would entail a wrong hypothesis.

5 RDTE Evaluation

We elicit sufficiency judgments over the 1000 gold-annotated RDTE items using a series of methods based on the RDTE protocol and existing methods for judging decompositional entailment. We test them on the gold sufficiency scores converted to binary labels by assigning 4 and 5s to positive entailment. Factuality, relevance, and redundancy predictions were not directly evaluated.

GPT Methods We test the following prompt-based methods, using both GPT-4 and ChatGPT.[6] All prompts can be found in the appendix.

- An in-context learning (**ICL**; Dong et al., 2023) prompt containing the RDTE annotation rubric and 4 example batches (10-15 decompositions per hypothesis) and a **Zero-Shot** prompt containing only the RDTE rubric.
- The prompts used by Clark et al. (2023) to evaluate models on the **BaRDa** dataset. These are separate prompts for the entailment and premise factuality judgments (see Figure 10).

Existing Methods We test various RTE models from recent entailment tree-based reasoners: the T5-based **Entailer-11B** (Tafjord et al., 2022), the similar **NELLIE-3B** model (Weir et al., 2024), as well as the fine-tuned RoBERTa classifier used by the NELLIE system as an entailment filter. We also take an off-the-shelf (**OTS**) RoBERTa finetuned for NLI using a suite of 600 datasets[7] that includes QASC (Khot et al., 2020) and ARC. All these methods predict binary entailment judgments.

Knowledge Distillation on Silver Data Search algorithms like TREEWISE make hundreds of calls to NLI models for every question, making large, slow models like GPT-4 unrealistic for use as an entailment filter. Instead, we explore using knowledge distillation to train smaller student models to imitate GPT -4's performance. We extracted 24K RDTE judgments from GPT-4 over 2.3K hypotheses in each of the ARC and Hotpot domains. We release these large silver datasets for future work.[8]

We used the silver data to fine-tune (A) the RoBERTa model used in the NELLIE engine, and (B) ChatGPT using OpenAI's fine-tuning API. We trained the former to predict the binarized silver sufficiency judgments and the latter to replicate GPT-4's exact answers to the RDTE prompt. For token efficiency, the GPT-4 teacher and ChatGPT student perform classification over a batch of decompositions for a single hypothesis in each prompt.[9] To collect binary predictions from these RDTE-prompted models, we assign 5s to a positive label prediction and ≤ 4 to a negative.

5.1 RDTE Results

Our metric is **F-score** ($\beta = 0.5$), putting double precedence on precision over recall. Precision is

[6] `gpt-4-0613` and `gpt-3.5-turbo-1106`

[7] huggingface.co/sileod/deberta-v3-large-tasksource-nli

[8] GPT-4 obtains a .61 and .44 Spearman ρ with humans on the ARC and HotpotQA RDTE splits, respectively.

[9] Batching barely affects performance on RDTE (Table 3).

	ARC	Hotpot
Prompted Methods		
GPT-4 (RDTE ICL)	**59**	**53**
GPT-4 (RDTE Zero-Shot)	58	49
GPT-4 (BaRDa)	44	48
ChatGPT (RDTE ICL)	36	35
ChatGPT (RDTE Zero-Shot)	40	38†
ChatGPT (BaRDa)	43†	34
T5 and Cross Encoders		
Entailer-11B	48	38
NELLIE-3B	43	36
NELLIE RoBERTa Filter	37	36
OTS NLI RoBERTa	45‡	44‡
Knowledge Distillation		
ChatGPT	48 (+5)†	51(+13)†
RoBERTa	**66** (+21)‡	**56**(+12)‡

Table 1: RDTE entailment results ($F_{0.5}$ score). We find that RDTE prompting of GPT-4 greatly outperforms existing approaches to compositional entailment, including BaRDa prompting and fine-tuned approaches. Knowledge disillation from GPT-4 (RDTE Zero-Shot) improves student models by 5-21 points, at times outperforming GPT-4 itself. †/‡ denote the student models and the best performing analogous non-distilled models.

particularly crucial for our needs: false positives in a backward-chaining search can quickly create error propagation, wasted time on invalid search branches, and worse trees. In contrast, while recall is important, it is less catastrophic to overfilter valid decompositions than to underfilter bad ones.

Table 1 shows $F_{0.5}$ performance by models on the RDTE dataset. We also display precision and recall in Table 3. We observe that **no model cracks 70**, suggesting room for future improvement. Overall we find the **RDTE-oriented prompts to GPT-4 outperform all existing methods for compositional entailment** by 10 or more points.[10]

We find that knowledge distillation proves effective on RDTE: the student ChatGPT models improve 5-13% over the closest ChatGPT methods, while the **fine-tuned RoBERTa student ultimately outperforms the teacher GPT-4 method**. Table 3 shows that this is due to higher precision (68 to 55) at the expense of recall (57 to 90). The RoBERTa student is the most precise classifier tested, and the ChatGPT student is the next-highest among non-GPT-4 models; however, both show much lower recall than the others tested.

6 TreeWise

To illustrate the impact that an RDTE-based entailment model can have on systematic reasoning, we apply it as a module in a new, state-of-the-art entailment engine called TreeWise: **T**extual **R**easoning **E**ngine with **E**nriched **W**ays to **I**ntelligently **S**earch for **E**ntailment. TreeWise builds upon the backward chaining entailment tree search framework first introduced by Nellie (Weir et al., 2024). The core functionality of TreeWise is to answer the question, **"is NL hypothesis H compositionally entailed by a corpus of documents C in the context of a question Q?"** Illustrated in Figure 4, the search attempts to ground the hypothesis via recursive decomposition into premises entailed by passages in a verified corpus such as Wikipedia. Implemented in Prolog, the algorithm follows a breadth-first strategy, performing the following at each recursive hypothesis H:

1. **Retrieve** a set of s support documents from a corpus like Wikipedia that are likely to contain information relevant to the hypothesis. Check if any of these documents entails H using a series of single-premise entailment classifiers. If so, H is proved. See §D for implementation details.
2. **Check whether H is a paraphrase** of a previously considered hypothesis H' using SBERT (Reimers and Gurevych, 2019a). If so, we associate it with the proof branches of H'.
3. **Forward generate** i inferences from the support documents via an instruction-tuned LLM.

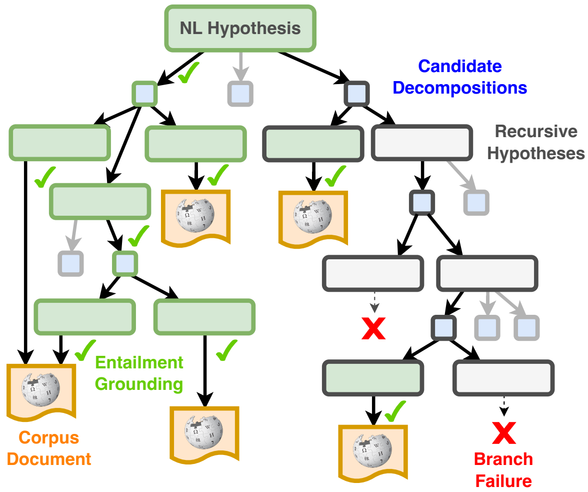


Figure 4: TreeWise generates many premise decompositions of a hypothesis and checks whether any candidates are valid entailments. Premises are then recursively decomposed until it finds any tree(s) fully grounded in one or more documents from a corpus like Wikipedia. Statements entailed by documents are generated via *forward chaining*, while the rest of the search is *backward*. Many decompositions end up untraversed due to the search budget or nonentailment.

[10] Table 1 shows GPT-4's score when thresholding on **5**, not **4**, which we found to score lower (see Table 3).

4. **Decompose** H into a set of d candidate decompositions using a series of prompts to the LLM. We use a diverse set of prompts to propose candidates that encourage using the i inferences.
5. **Condense the decompositions** by deduplicating semantically equivalent candidates.
6. **Filter the decompositions** using a series of decompositional entailment classifiers to weed out those that would not logically support H.
7. **Recur on the premises** to continue the search.

This process is repeated until either a user-specified expansion budget is exhausted, or a proof is found and no search branches remain that could yield a higher score. The system is designed to be amenable to a costly API; it batches together decompositions of a single hypothesis into one prompt and then asks it to score them all at once.

TREEWISE represents an overhaul of the T5-based NELLIE system, incorporating a variety of prompting methods, forward chaining, and the capacity to reason over long passages. We discuss the search logic and architectural comparison in §A.

7 TREEWISE Experiments

The distillation framework suggests a way forward for improving entailment engines in new domains: first extract non-optimized reasoning traces from the engine, annotate them under the RDTE protocol with GPT-4 (which is itself too slow and expensive to use in the engine), train student models on the silver data, and then substitute the student as an entailment classifier in the engine. The experiments below emulate this scenario and show that the resulting engines improve on both QA and on generated tree quality, highlighting the effectiveness of applying RDTE-trained distillation student models to entailment tree creation algorithms.

Entailment Tree QA Task Definition We pursue a version of evidence-grounded, explainable QA in which models are right for the right reasons. Given corpus C, multiple-choice question Q and hypotheses $h_1, \ldots, h_m$ corresponding to answer options $a_1, \ldots, a_m$, we task models to predict correct answer a_i^* and generate entailment tree T_{h_i} rooted at h_i such that (1) the T_{h_i}'s leaves are in $C \cup Q$ and (2) the tree's steps explain how each parent follows compositionally from its children. Models are evaluated on the selection of a_i^* and the correctness of the reasoning in T_{h_i}.

7.1 QA Evaluation

Datasets We evaluate TREEWISE and a series of entailment tree-creating baselines on the two QA datasets used to construct RDTE: 340 test questions from EntailmentBank (**EBQA**; Dalvi et al., 2021), and 419 **HotpotQA** questions recast as multiple choice by using GPT-4 to generate incorrect answer options.

As TREEWISE can flexibly hook up to different types of knowledge sources, we evaluate it on EBQA using two different scenarios: (1) EBQA using the clean factbase WorldTree (Xie et al., 2020) as the knowledge source, and (2) EBQA using an index over English Wikipedia as the knowledge source. For HotpotQA, we only use that task's specific Wikipedia index as the knowledge source.

Metrics We take a two-pronged approach to evaluating systems: they should produce both strong QA accuracy **and** coherent and logically sound entailment trees explaining the chosen answer. To evaluate the latter, we introduce a new model-based **tree integrity score**. We use GPT-4 to score each of a tree's component entailment steps under the RDTE protocol. Following the intuition that an argument is only as strong as its weakest link, we take the minimum score as the tree's overall score.

Baselines We compare a version of TREEWISE using the RDTE-trained student ChatGPT, and RoBERTa models to an identical version that uses an ICL-prompted ChatGPT and the RoBERTa model used by NELLIE, which is trained on non-RDTE entailment data. We use ChatGPT as the decomposition generator and QA2D model for TREEWISE, meaning it does not use GPT-4 for anything at test time. We also evaluate a pair of greedy baselines for entailment tree creation. These baselines are designed to mimic TREEWISE's behavior in a simplified manner without the systematic search:

- An **end-to-end** tree generator that retrieves one set of facts and then uses ChatGPT to decode an entailment tree in one fell swoop.
- A **stepwise** tree generator that iteratively retrieves support facts and then decodes one step of the entailment tree until the tree is fully grounded or a maximum number of steps (10) is reached.

Pseudocode for these is provided in §F. Each process is repeated 5 times, yielding 5 different candidate trees for each answer choice, then fed to the **tree integrity** scorer using the student ChatGPT

Method	W/ Silver?	EBQA on WorldTree QA	EBQA on WorldTree Tree Integrity	EBQA on Wikipedia QA	EBQA on Wikipedia Tree Integrity	HotpotQA on Wikipedia QA	HotpotQA on Wikipedia Tree Integrity
TREEWISE	Yes	**79.2**	**75.2**	**73.2**	**74.7**	**51.3**	**66.6**
	No	72.8	71.6	69.0	71.9	46.1	62.9
Stepwise Generator	Yes	66.4	69.5	56.8	68.3	47.5	58.8
	No	63.7	68.1	51.5	64.7	49.4	58.6
End-to-End Generator	Yes	68.7	66.4	57.6	66.4	48.5	59.1
	No	68.7	66.1	55.1	64.0	43.0	57.7

Table 2: QA and tree integrity score for tree-generating approaches with vs without silver knowledge distillation.

to score each tree. We take the highest-scoring tree and corresponding answer as the final output. We compare these to non-RDTE versions where the student ChatGPT is replaced by regular ChatGPT.

Results Table 2 shows QA results for these methods. We observe that for all methods, the tree integrity score increases when using the knowledge-distilled student. For all but 1 baseline on HotpotQA, QA accuracy increases by 1 to 7 points. We observe that TREEWISE achieves the highest QA and integrity scores, while the stepwise outperforms the end-to-end generator under integrity but vice versa for two QA scenarios. See Figure 15 and 16 for example trees generated by TREEWISE using Wikipedia as its knowledge source.

8 Conclusion

The trustworthy application of LLMs to complex reasoning tasks critically depends not only on accurate responses, but also accurate justifications. To date, research in entailment tree reasoning has largely focused on measuring the former.

This work develops a protocol based on works in informal logic for the assessment of compositional entailment, the core reasoning task undergirding entailment trees. We employed this rubric-based protocol to annotate RDTE, a high-quality dataset of judged compositional entailments, along with tens of thousands of automatically scored items by GPT-4 under the same instructions in multiple domains. We demonstrated that RDTE supports state-of-the-art results with a novel system for evidence-grounded entailment tree generation. This pipeline, which involves knowledge distillation of student models using careful entailment prompting, can be readily applied to new task domains to make entailment-based reasoners more precise task-specific problem solvers.

The combination of our rubric, manual annotations, GPT-4 derived data, and this new system that we call TREEWISE represents a significant advance in building trustworthy AI systems capable of not only solving complex reasoning tasks but providing correct justifications along with their answers.

Limitations

The RDTE dataset is a high-quality set of 1000 decompositions across two specific QA domains. As argument sufficiency is a domain-dependent notion, we extensively discussed what constituted validity in the two different tasks. Applying the RDTE protocol to new domains will likely also merit careful consideration of how the various facets of the task manifest differently for different types of questions.

A system such as TREEWISE does not carry direct risks towards others; however, since most automated reasoning systems can exacerbate biases already existing within language and culture, we recognize that our reasoning algorithm has an inherent potential to cause damage to certain groups and identities.

Acknowledgements

The authors thank Marc Marone and Rachel Wicks for their feedback on earlier drafts. We also thank Abbey Coogan for contributions on figures. Nathaniel Weir is supported by an NSF GRFP. This work has been supported in part by the U.S. National Science Foundation under grant No.2204926.

References

Gabor Angeli and Christopher D. Manning. 2014. NaturalLI: Natural logic inference for common sense reasoning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 534–545. Association for Computational Linguistics.

J Anthony Blair. 2012. Relevance, acceptability and sufficiency today. *Groundwork in the Theory of Argumentation: Selected Papers of J. Anthony Blair*, pages 87–100.

Johan Bos. 2014. Recognizing textual entailment and computational semantics. In *Computing Meaning: Volume 4*, pages 89–105. Springer.

Kaj Bostrom, Zayne Sprague, Swarat Chaudhuri, and Greg Durrett. 2022. Natural language deduction through search over statement compositions. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4871–4883. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642. Association for Computational Linguistics.

Tongfei Chen, Zhengping Jiang, Adam Poliak, Keisuke Sakaguchi, and Benjamin Van Durme. 2020. Uncertain natural language inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8772–8779. Association for Computational Linguistics.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Peter Clark, Bhavana Dalvi Mishra, and Oyvind Tafjord. 2023. BaRDa: A belief and reasoning dataset that separates factual accuracy and reasoning ability. *arXiv preprint arXiv:2312.07527*.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*.

Bhavana Dalvi, Peter Jansen, Oyvind Tafjord, Zhengnan Xie, Hannah Smith, Leighanna Pipatanangkura, and Peter Clark. 2021. Explaining answers with entailment trees. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7358–7370. Association for Computational Linguistics.

Dorottya Demszky, Kelvin Guu, and Percy Liang. 2018. Transforming question answering datasets into natural language inference datasets. *arXiv preprint arXiv:1809.02922*.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. 2023. A survey for in-context learning. *ArXiv*, abs/2301.00234.

Max Glockner, Ieva Staliunaite, James Thorne, Gisela Vallejo, Andreas Vlachos, and Iryna Gurevych. 2021. Ambifc: Fact-checking ambiguous claims with evidence. *Transactions of the Association for Computational Linguistics*, 12:1–18.

Leo Groarke. 2022. Informal Logic. In Edward N. Zalta and Uri Nodelman, editors, *The Stanford Encyclopedia of Philosophy*, winter 2022 edition. Metaphysics Research Lab, Stanford University.

Reto Gubelmann, Ioannis Katis, Christina Marianne Niklaus, and Siegfried Handschuh. 2023. Capturing the varieties of natural language inference: A systematic survey of existing datasets and two novel benchmarks. *Journal of Logic, Language and Information*, pages 1–28.

Ruixin Hong, Hongming Zhang, Xintong Yu, and Changshui Zhang. 2022. METGEN: A module-based entailment tree generation framework for answer explanation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1887–1905. Association for Computational Linguistics.

Ruixin Hong, Hongming Zhang, Hong Zhao, Dong Yu, and Changshui Zhang. 2023. Faithful question answering with Monte-Carlo planning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3944–3965. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Zhijing Jin, Abhinav Lalwani, Tejas Vaidhya, Xiaoyu Shen, Yiwen Ding, Zhiheng Lyu, Mrinmaya Sachan, Rada Mihalcea, and Bernhard Schoelkopf. 2022. Logical fallacy detection. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 7180–7198. Association for Computational Linguistics.

Ralph H. Johnson and J. Anthony Blair. 1977. Logical self-defense.

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.

Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021)*, pages 2356–2362.

Christopher D Manning. 2006. Local textual inference: it's hard to circumscribe, but you know it when you see it–and nlp needs it.

OpenAI. 2023. Gpt-4 technical report. *ArXiv*, abs/2303.08774.

Raquel Mochales Palau and Marie-Francine Moens. 2009. Argumentation mining: the detection, classification and structure of arguments in text. In *Proceedings of the 12th International Conference on Artificial Intelligence and Law*, ICAIL '09, page 98–107, New York, NY, USA. Association for Computing Machinery.

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent disagreements in human textual inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694.

Nils Reimers and Iryna Gurevych. 2019a. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019b. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Danilo Ribeiro, Shen Wang, Xiaofei Ma, Rui Dong, Xiaokai Wei, Henghui Zhu, Xinchi Chen, Peng Xu, Zhiheng Huang, Andrew Arnold, and Dan Roth. 2022. Entailment tree explanations via iterative retrieval-generation reasoner. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 465–475, Seattle, United States. Association for Computational Linguistics.

Danilo Neves Ribeiro, Shen Wang, Xiaofei Ma, Henghui Zhu, Rui Dong, Deguang Kong, Juliette Burger, Anjelica Ramos, zhiheng huang, William Yang Wang, George Karypis, Bing Xiang, and Dan Roth. 2023. STREET: A MULTI-TASK STRUCTURED REASONING AND EXPLANATION BENCHMARK. In *The Eleventh International Conference on Learning Representations*.

Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at trec-3. *Nist Special Publication Sp*, 109:109.

Kate Sanders, Nathaniel Weir, and Benjamin Van Durme. 2024. Tv-trees: Multimodal entailment trees for neuro-symbolic video reasoning. *EMNLP*.

Christian Stab and Iryna Gurevych. 2017. Recognizing insufficiently supported arguments in argumentative essays. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 980–990. Association for Computational Linguistics.

Justin Sulik, Jeroen van Paridon, and Gary Lupyan. 2021. Explanations in the wild. *Cognition*, 237.

Oyvind Tafjord, Bhavana Dalvi Mishra, and Peter Clark. 2022. Entailer: Answering questions with faithful and truthful chains of reasoning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2078–2093. Association for Computational Linguistics.

Chenhao Tan. 2022. On the diversity and limits of human explanations. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2173–2188. Association for Computational Linguistics.

Marco Valentino, Ian Pratt-Hartmann, and André Freitas. 2021. Do natural language explanations represent valid logical arguments? verifying entailment in explainable NLI gold standards. In *Proceedings of the 14th International Conference on Computational Semantics (IWCS)*, pages 76–86. Association for Computational Linguistics.

Nathaniel Weir, Peter Clark, and Benjamin Van Durme. 2024. NELLIE: A neuro-symbolic inference engine for grounded, compositional, and explainable reasoning. In *Proceedings of IJCAI*.

Sarah Wiegreffe and Ana Marasovic. 2021. Teach me to explain: A review of datasets for explainable natural language processing. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

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 Twelfth Language Resources and Evaluation Conference*, pages 5456–5473. European Language Resources Association.

Kaiyu Yang, Jia Deng, and Danqi Chen. 2022. Generating natural language proofs with verifier-guided search. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 89–105. Association for Computational Linguistics.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering.

In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380. Association for Computational Linguistics.

Sheng Zhang, Rachel Rudinger, Kevin Duh, and Benjamin Van Durme. 2017. Ordinal common-sense inference. *Transactions of the Association for Computational Linguistics*, 5:379–395.

A Contrasting NELLIE with TREEWISE

NELLIE (Weir et al., 2024) is a T5-based compositional entailment engine that shows high performance on Science QA by checking whether answer hypotheses are compositionally entailed by the WorldTree knowledge base (Xie et al., 2020). Its search algorithm suffers from the primary drawbacks that (1) it requires a clean corpus of knowledge sentences, which is not always available for a particular problem domain, and (2) its various modules all rely on different in-domain training datasets not typically available for new tasks.

In this section, we introduce TREEWISE, an evolution of NELLIE based on instruction-tuned LLMs like ChatGPT and GPT-4. TREEWISE introduces a series of improvements to the engine's search algorithm that allow it to handle novel domains (1) with noisier knowledge sources like an index over Wikipedia passages common to state-of-the-art question-answering systems and (2) without module-specific training data. In this way, TREEWISE can answer whether a hypothesis from a novel QA dataset is compositionally entailed by Wikipedia.

A.1 Search Logic

We refer readers to Weir et al. (2024) for an overview of the original NELLIE search algorithm for compositionally grounding a hypothesis in a corpus. Their search generally follows a breadth-first search across candidate decompositions by following 3 Prolog rules:[11]

1. *Fact Unification*
   $\textsc{Prove}(h) \Leftarrow \textsc{Retrieve}(h^+, f^-) \wedge \textsc{Entails}(f, h)$

2. *Two Premise Rule Generation*
   $\textsc{Prove}(h) \Leftarrow \textsc{RuleGen}(h^+, f_1^-, f_2^-) \wedge$
   $\textsc{Entails}([f_1, f_2], h) \wedge \textsc{Prove}(f_1) \wedge \textsc{Prove}(f_2)$

3. *Retrieved First Premise Rule Generation*
   $\textsc{Prove}(h) \Leftarrow \textsc{Retrieve}(h^+, f_1^-) \wedge$
   $\textsc{RuleGen}(h^+, f_1^+, f_2^-) \wedge \textsc{Entails}([f_1, f_2], h) \wedge$
   $\textsc{Prove}(f_2)$

We make the following observations:

(a) Rules 2/3 constrain the search to only binary conjunctions; allowing 3 or 4 might allow for added flexibility.

(b) The LM-calling RULEGEN predicate only ever accepts 0 or 1 support facts; conditioning on multiple retrieved candidate facts might produce higher quality and/or more groundable decompositions.

(c) Rule 3 assumes that items returned by RETRIEVE (f_1^-) are of the same type as the premises that constitute an entailment: in NELLIE's case, simple evidential sentences like "birds can fly." If TREEWISE's corpus contains noisier knowledge, e.g. Wikipedia paragraphs, then this assumption becomes problematic.

(d) Rules 2/3 assume that any unique f_1, f_2, or h is semantically distinct from other statements considered during the search. This implies that recursively calling PROVE on a new statement is never a waste of time. In practice, however, NELLIE frequently considers hypotheses that are paraphrases of each other. This risks wasting substantial search time, especially if neither the hypothesis nor its paraphrase will ever be grounded.

To address these issues, we make the following modifications, replacing rules 2 and 3 with rules 4, 2*, 3*. The latter two rules are only executed if 4 does not succeed. Bolded symbols denote string lists.

4. *Paraphrase Unification*
   $\textsc{Prove}(h) \Leftarrow \textsc{Expanded}(g^-) \wedge$
   $\textsc{Paraphrase}(h^+, g^+) \wedge \textsc{Prove}(g)$

2*. *Non-Conditioned Rule Generation*
   $\textsc{Prove}(h) \Leftarrow \textsc{RuleGen}(h^+, [], \mathbf{f}) \wedge$
   $\textsc{Entails}(\mathbf{f}, h) \wedge \textsc{MapList}(\mathbf{f}^+, \textsc{Prove})$[12]

3*. *Retrieval-Conditioned Rule Generation*
   $\textsc{Prove}(h) \Leftarrow \textsc{Retrieve}(h^+, \mathbf{s}^-) \wedge$

[11]We postulate that Prolog terms are evaluated in the sequence they are read, as is typical in most executors.

[12]https://www.swi-prolog.org/pldoc/man?predicate=maplist/2

	Training Data	RDTE–ARC			RDTE–Hotpot			BaRDa		
		Pr	**Re**	**F$_{0.5}$**	**Pr**	**Re**	**F$_{0.5}$**	**Pr**	**Re**	**F$_{0.5}$**
Itemwise GPT Methods										
GPT-4 (ICL)	RDTE Directions + 4 Exemplars	55	90	**59**	**49**	74	**53**	**92**	46	76
(w/ Threshold 4)		49	**100**	55	45	86	50	90	58	81
GPT-4 (Zero-Shot)	RDTE Directions Only	59	57	58	47	60	49	91	31	66
(w/ Threshold 4)		39	**100**	45	35	91	40	83	70	80
GPT-4 (BaRDa)	BaRDa Directions + 10 Exemplars	40	72	44	43	95	48	82	85	**83**
ChatGPT (ICL)	RDTE Directions + 4 Exemplars	31	97	36	30	95	35	69	91	72
ChatGPT (Zero-Shot)	RDTE Directions Only	36	64	40	39	33	38	72	15	41
ChatGPT (BaRDa)	BaRDa Directions + 10 Exemplars	38	94	43	30	79	34	73	84	75
Batched GPT Methods										
GPT-4 (ICL)		52	79	56	52	62	54		(N/A)	
GPT-4 (Zero-Shot)		52	64	54	52	51	52		(N/A)	
T5 and Cross Encoders										
Entailer-11B	EntailmentBank + Entailer	45	68	48	33	90	38	76	83	77
NELLIE-3B	EntailmentBank + Entailer + QASC	39	74	43	31	95	36	72	94	75
NELLIE RoBERTa Filter	EntailmentBank + Entailer + QASC	33	76	37	32	95	36	68	**95**	72
OTS NLI RoBERTa	Various NLI incl QASC	42	68	45	39	85	44	76	90	79
Knowledge Distillation Students										
ChatGPT	GPT-4 (Zero-Shot) Silver Data	46	58	48	52	49	51	82	55	75
RoBERTa	GPT-4 (Zero-Shot) Silver Data	**68**	57	**66**	**56**	56	**56**	83	47	72

Table 3: Entailment results by various approaches on the RDTE and BaRDa datasets. Batching decompositions by their shared hypothesis does not drastically impact performance by GPT-4. Batching was not possible for BaRDa, which has one item per hypothesis.

$$\textsc{InferenceGen}(h^+, \mathbf{s}^+, \mathbf{i}^-) \wedge$$
$$\textsc{RuleGen}(h^+, \mathbf{i}^+, \mathbf{f}^-) \wedge \textsc{Entails}(\mathbf{f}, h) \wedge$$
$$\textsc{MapList}(\mathbf{f}^+, \textsc{Prove})$$

The novel predicate EXPANDED is a non-deterministic function that evaluates to true iff g^- was a previous input to RULEGEN during the search. The predicate PARAPHRASE uses SBERT (Reimers and Gurevych, 2019a) cosine similarity to identify paraphrases. The predicate INFERENCEGEN is a **forward chaining** inference generator that receives a list of support items (facts, passages, or otherwise) and returns a list of sentential premises likely entailed by the items that might be helpful to prove h. The new RULEGEN now accepts an arbitrary list of candidate premises ($\mathbf{i}$) and returns an arbitrary-length decomposition ($\mathbf{f}$). As a result, the premises in the decomposition might not have appeared in $\mathbf{i}$ or $\mathbf{s}$; this adds flexibility to the generator but also means that the algorithm has to call PROVE on all generated premises. If $f \in \mathbf{f}$ does appear in $\mathbf{i}$ or $\mathbf{s}$, it is likely immediately grounded via fact unification (rule 1).

A.2 Prompt-based Modules

With the introduction of instruction-tuned LLMs like ChatGPT and GPT-4, we find it no longer necessary to train certain NELLIE models via supervised learning on in-domain datasets. We replace the modules for query declarativization (Demszky et al., 2018), decomposition generation, and 1- and multi-premise entailment filtering with a mixture of in-context learning and zero-shot instruction prompts to a GPT model. This includes the predicates RULEGEN and ENTAILS above. All prompts can be found in the appendix and will be released with the rest of the codebase.

A.3 Improved Reasoning Generators

A key improvement of TREEWISE over its predecessor is the redesigned approach to generating and filtering candidate decompositions for decompositional entailment. NELLIE's original decomposition generator is a T5-based model trained via supervised learning on data from a specific domain. It is difficult to (A) adapt to new domains without retraining and (B) convey to the model that decompositions serve a reasoning-related purpose. With the introduction of instruction-tuned LLMs, this becomes more straightforward. For TREEWISE, we replace the T5-based generator with a diverse series of prompts for generating ad-hoc decompositions of a hypothesis in any domain:

- A **fact-conditioned** prompt (Figure 13) that receives a list of forward-chaining inferences derived from support documents using a separate prompt (Figure 11) and returns a list of candidate decompositions.

- A **follow-up generation** prompt that receives the output of the previous prompt and an instruction to revise them to better fit the given hypothesis and question
- An **in-context learning** prompt (Figure 12) dynamically constructed by retrieving exemplars from a fixed set of training items using BM25. We use as our training set the gold-annotated inferences from EntailmentBank (Dalvi et al., 2021).

Together these prompts populate an initial horizon of candidate decompositions to be condensed, checked for semantic equivalence, and subsequently filtered for argument validity.

B RDTE Annotation Instructions

Figure 5 shows the rubric and condition lists for evaluating premise-specific facts (relevance, factuality, redundancy). Figure 6 shows the rubric and condition list for annotating sufficiency. These are ARC-specific lists; we constructed a similar but slightly different version for HotpotQA. We will release both rubrics with the dataset.

B.1 Reasoning Filters

Our RDTE-oriented prompting strategy discussed in the main body of the paper is shown in Figure 7, 8, and 9. The zero-shot variant contains the same directions but no exemplars. We use a separate set of exemplars for Hotpot vs ARC.

In addition to this improvement at multi-premise compositional NLI, we also implement a single-premise/passage entailment rubric for use by TREEWISE and when computing our tree integrity score. This prompt rates entailment on an ordinal 1-5 scale analogous to the RDTE protocol for compositional entailment; the prompt is shown in Figure 14.

C Example TREEWISE Outputs

Figure 15 and Figure 16 depict example outputs by the TREEWISE tree search algorithm when hooked up to Wikipedia as the knowledge source for answering ARC and HotpotQA questions.

D Retrieval Index

We build retrieval indexes for HotpotQA and Wikipedia using the `pyserini` package (Lin et al., 2021). We index all the data in HotpotQA as given in the original paper and index the 2021-01-20 version of Wikipedia with 100 word chunks. We use the BM25 algorithm (Robertson et al., 1995) for first stage retrieval and rerank using SentenceTransformer's `ms-marco-MiniLM-L-12-v2` (Reimers and Gurevych, 2019b).

We retrieve the top 1000 documents and rerank and return the top 30 candidate support facts per hypothesis using as our retrieval query the concatenated question and hypothesis.

E Hyperparameters

We train the ChatGPT student for 5 epochs using the OpenAI API, and the RoBERTa student for 10 epochs using the SentenceTransformers library (Reimers and Gurevych, 2019a). We trained separate student models on 18.3K ARC decompositions and on 20.4K Hotpot ones.

In the TREEWISE algorithm, we prompt the decomposition generators to propose 10 decompositions each, resulting in 40 candidate decompositions per hypothesis. To improve the initial search horizon, we double this number at depth 0 only. We prompt the forward-chaining prompt to produce 30 inferences entailed by retrieved documents. We use temperature=.2 sampling for all entailment filters and temperature=1 for generating decompositions.

We set the expansion budget to 80 nodes and the filter entailment filter threshold to 0.6. We define paraphrases as having SBERT cosine similarity of 0.9 or higher.

F Baselines

Algorithm 1 depicts the pseudocode for the end-to-end tree generation baseline. Algorithm 2 depicts the stepwise version.

Factuality	How factual is the fact? 1 is completely false, 5 is completely true.
Relevancy	How relevant is the fact to helping decompose the conclusion? An irrelevant fact is one that either does not address a key aspect of the conclusion, introduces irrelevant information, or is otherwise unnecessary should be scored lower in relevance. 1 is completely irrelevant, 5 is completely relevant. If the fact is situationally relevant to the conclusion, but contradicts it, you should still score it 5.
Redundancy	Does the fact introduce new information that is not already contained in other facts in the decomposition? 1 is completely redundant, e.g., the third fact completely restates the first one, and 5 is completely new information. Sometimes, one of the facts directly restates the entire conclusion by itself, which **should be marked with the checkmark only and should not affect the numerical score**. Otherwise, facts including information included in the conclusion are acceptable and strictly necessary.

Factuality Questions to Ask Yourself	
Is a fact ambiguously grounded in the question context in a way that does *not* affect the reasoning? (e.g. the fact "two sticks are rubbed together" in the question "what is an example of a force producing heat? (A) two sticks rubbed together, . . . ")	This is acceptable and can be 5/5 factuality
Is a fact true in nearly all cases except extreme ones that don't pertain to the question?	(Yes = 5 factuality)
Relevancy Questions to Ask Yourself	
Is a fact not on topic? ("on topic" is defined as containing nouns or entities that appear in the hypothesis)	(Yes = 1 relevancy)
Does there not exist some (potentially over-pedantic) decomposition in which the given fact would be necessary to complete the entailment?	(Yes = max 2 relevancy)
Would removing an on-topic fact ***in isolation*** **not** change the extent to which the conclusion is supported?	(Yes = 2 relevancy)
Would removing an on-topic fact in isolation **minimally** change the extent to which the conclusion is supported?	(Yes = 3 relevancy)
Redundancy Questions to Ask Yourself	
Is a fact a paraphrase of another fact?	(Yes = 1 redundancy for second fact)
Does a given fact add entailment in isolation, but if you removed the fact ***conditioned on the rest of the facts***, it **would not** change the extent to which the conclusion is supported?	(Yes = max 2 redundancy)
Does a fact restate information in the **question text** (*not* the conclusion)?	This is acceptable. Only check for restatements of other facts and/or the conclusion. Restatement of the question text, especially to cite evidence, is fine.

Figure 5: RDTE annotation guidelines for premise-specific qualia in ARC.

1 (Malformed or Nonsensical)	Completely incorrect logic, or contains a fact that contradicts the conclusion, or malformed facts (not complete sentences), or inter-fact pronoun references (e.g. "this" or "that" or "such").
2 (Poor)	Any two of the following: (1) some nontrivial amount of redundancy, (2) one irrelevant fact (2/5 or lower), (3) missing/implicit information that makes deducing the conclusion impossible without a substantial leap in logic. Would not convince a human of the conclusion.
3 (Moderately Correct)	Generally coherent and correct, but there is some significant flaw. E.g., **one of the facts is untrue** but if it was true the proof would be correct.
4 (Mostly Correct)	Slight redundancy or missing/implicit information, but **not to the point that it should substantially impact a human** performing the reasoning.
5 (Perfect)	Completely correct and sound decomposition. No redundancy and no missing/implicit information. No ambiguous language. Addresses all conditions required to infer the conclusion. Does not leave anything implicit.

Questions to Ask Yourself	
Are any of the premises not well-formed? (no fragments, no 1 sentence that was split into two non-sentence parts)	(Yes = 1)
Do the premises together or individually *contradict* the conclusion instead of supporting it?	(Yes = 1)
Are there between-premise pronoun references ('this', 'that', 'such') whose antecedent would be ambiguous without the other premises?	(Yes = 1)
Are there any conjunctive adverbs like "therefore" or "thus"?	(Yes = 1)
Are all premises irrelevant, off-topic, or not contributing any correct logic? (e.g. all 1's for relevance, or removing all of them would not change the extent of the entailment)	(Yes = 1)
Does any premise essentially restate the conclusion without adding/removing any information?	(Yes = max 2)
Are there at least two of the following? (1) redundant fact, (2) untrue fact, (3) irrelevant fact, (4) missing information	(Yes = max 2)
Is the conclusion assuming an effect that isn't directly linked to the cause, or overlooking more immediate effects?	(Yes = max 2)
If you removed all redundant or irrelevant (3/5 or lower) facts, would there be only one fact remaining **and not full entailment**?	(Yes = max 2)
Is there any amount of logical error present?	(Yes = max 2)
Did you give any fact a 2/5 or lower for factuality or relevance?	(Yes = max 3)
Does proving one premise amount to proving all of the others?	(Yes = max 3)
Are the premises all factual and relevant, but there is a part of the conclusion (e.g. something non-common-sense or a thing that a 10-year-old would **not** intuit in the context of the question) that is not stated or explained?	(Yes = max 3)
If you removed all redundant or irrelevant (3/5 or lower) facts, would there be only one fact remaining?	(Yes = max 3)
Are the premises all evidence statements entailed by the question context and nothing else?	(Yes = **likely** max 3)
Is one of the facts not true, but if it were then it'd be a perfect entailment?	(Yes = 3)
Did you give any fact a 3/5 or lower for redundancy, factuality, or relevance?	(Yes = max 4)
Are there two separate arguments being partially/mostly made to support the hypothesis, but one/both is missing some implicit premises?	(Yes = max 4)
Are the premises all factual and relevant, but the conclusion does not follow from the premises for a minor reason (e.g. a common sense-y fact that would have been inferred by a 10-year-old in the context of the question)	(Yes = max 4)
Do two of the facts perfectly entail the conclusion, but the third is essentially redundant?	(Yes = 4)
Is the conclusion irrelevant to the question, but the entailment supports the conclusion perfectly?	(Yes = 5)
Does the question ask something along the lines of "which is the best...?" and the entailment doesn't mention the other answer options?	Treat it like the "best" is not there
Do the premises not properly entail the conclusion for some other reason?	Reach out to us for clarification
Is one premise P1 an effective paraphrase of the hypothesis H, but another premise P2 serves as lexical grounding between P1 and H?	Ignore the paraphrase = max 2 rule

Figure 6: RDTE annotation guidelines for annotating sufficiency in ARC.

RDTE judgment prompt:

```
You are a reasoning system that searches for proofs of a hypothesis by recursively decomposing it into simpler premises.

Given a question and a hypothesis, you give a list of possible decompositions of the hypothesis into premises such that proving
      the list of premises would amount to proving the hypothesis through compositional entailment. The hypothesis might represent
       an answer to the question ,or it might represent a recursive query. However, many of the decompositions are incorrect, and
      you must identify which ones are correct and which ones are incorrect. For the following question, hypothesis and list of
      premise decompositions, score each decomposition according to the following rubrics:

You will first score each premise on a scale of 1 to 5 for each of the following qualia:
Factuality: How factual is the premise? 1 is completely false, 5 is completely true.
Relevance: How relevant is the premise to helping explain the hypothesis? 1 is completely irrelevant, 5 is completely relevant.
Redundancy: Does the premise introduce new information that is not already contained in other premises? 1 is completely redundant,
       i.e. completely restating another premise or the hypothesis, 5 is completely new information.

You will then judge whether the decomposition as a whole constitutes a complete inference, on a scale of 1 to 5 using the
      following rubric:
1 (malformed or nonsensical): Completely incorrect logic, or contains a premise that contradicts the hypothesis, or malformed
      instances, or inter-premise pronoun references (E.g. a "this" in premise 2 that refers to premise 1).)
2 (poor): Some nontrivial amount of redundancy, one irrelevant fact, and/or missing information. Would impact a human performing
      reasoning.
3 (moderately correct): Generally coherent and correct, but there is some significant flaw. (E.g., one of the facts is untrue but
      if it was true the proof would be correct.)
4 (mostly correct): Slight redundancy or missing information, but not to the point that it should substantially impact a human
      performing the reasoning.
5 (perfect): Completely correct and sound decomposition. No redundancy and no missing information. No ambiguous language.

You are renowned for your stringent eye. There should be minimal "information loss" between the hypothesis and the premises; you
      are looking for strict entailment. YOU RARELY GIVE A 5

Finally, you will provide an explanation for your judgment. Your explanation should justify any non-perfect scores you have given
      for factuality, relevance, and redundancy. In other words, explain why you gave a premise a certain score based on the
      information in the premise and its relation to the hypothesis.
For the complete inference score, explain why the conjunction of premises either does or does not amount to a complete and correct
       proof of the hypothesis. If there were issues with the complete inference, identify what information was missing or what
      logical errors were made. Your explanation should be clear and concise, providing valuable insight into your scoring process

Your output should be serialized json items, one per line, and nothing else.

QUESTION 1:
Which of the following items conducts electricity? (A) a lego brick, (B) a suit of armor, (C) a wooden table, (D) a T-shirt

HYPOTHESIS 1:
A suit of armor conducts electricity

DECOMPOSITIONS 1:
(1) a suit of armor is made of iron AND iron is a metal
(2) armor is made of metal AND metal conducts electricity
(3) armor cannot be punctured AND iron conducts electricity
(4) armor is made of iron AND iron is a metal AND metal conducts electricity
(5) armor is an object AND objects conduct electricity AND armor is an object that is made of metal
(6) a wooden table is made of wood AND wood conducts electricity
(7) conductivity is the degree to which a material conducts electricity AND conductivity is measured as the ratio of current
      density to the electric field that causes the flow of current.
(8) an item conducts electricity if the material that it is made of conducts electricity AND metal conducts electricity
(9) a suit of armor is made of iron AND iron is a metal AND conductivity is measured in Siemens per meter
(10) metal conducts electricity AND a suit of armor conducts electricity

JUDGEMENTS 1 (10 items):
{{"index": 1, "factuality": [4, 5], "relevance": [5, 5], "redundancy": [5, 5], "complete_inference": 2, "explanation": "The fact
      that armor is made of iron and iron is a metal does not necessarily mean that armor conducts electricity."}}
{{"index": 2, "factuality": [5, 5], "relevance": [5, 5], "redundancy": [5, 5], "complete_inference": 5, "explanation": "Properly
      identifies that armor is made of a type of material (metal) that conducts electricity."}}
{{"index": 3, "factuality": [5, 5], "relevance": [1, 5], "redundancy": [5, 5], "complete_inference": 2, "explanation": "The fact
      that armor cannot be punctured is irrelevant to whether armor conducts electricity."}}
{{"index": 4, "factuality": [3, 5, 5], "relevance": [5, 5, 5], "redundancy": [5, 5, 5], "complete_inference": 4, "explanation": "
      identifies that armor is made of a material (iron) that is a type (metal) that conducts electricity, but armor is not always
       made of iron."}}
{{"index": 5, "factuality": [5, 2, 5], "relevance": [3, 3, 5], "redundancy": [1, 5, 5], "complete_inference": 3, "explanation": "
      Not all objects conduct electricity, so the premise 'objects conduct electricity' is an overgeneralization."}}
{{"index": 6, "factuality": [5, 1], "relevance": [1, 2], "redundancy": [5, 5], "complete_inference": 2, "explanation": "The
      premises are about a wooden table, not referencing a suit of armor, and wood does not conduct electricity."}}
{{"index": 7, "factuality": [5, 5], "relevance": [1, 1], "redundancy": [5, 5], "complete_inference": 2, "explanation": "The
      premises are about the general concept of conductivity, not specifically about a suit of armor or metal."}}
{{"index": 8, "factuality": [5, 5], "relevance": [5, 5], "redundancy": [5, 5], "complete_inference": 3, "explanation": "Does not
      include that a suit of armor is made of metal"}}
{{"index": 9, "factuality": [5, 5, 5], "relevance": [5, 5, 1], "redundancy": [5, 5, 5], "complete_inference": 2, "explanation": "
      the measurement of conductivity in Siemens per meter is unnecessary to prove that a suit of armor conducts electricity."}}
{{"index": 10, "factuality": [5, 5], "relevance": [5, 5], "redundancy": [5, 1], "complete_inference": 2, "explanation": "Second
      premise restates the hypothesis"}}
```

Figure 7: Directions and in-context learning exemplar prompt used to extract RDTE judgments from instruction-tuned models. The zero-shot variant has the same first half but no examples.

RDTE judgment prompt (continued):

```
QUESTION 2:
Natural selection cannot occur without (A) competition for unlimited resources. (B) gradual warming of Earth. (C) genetic
      variation in species. (D) asexual reproduction in species.

HYPOTHESIS 2:
Natural selection cannot occur without competition for unlimited resources.

DECOMPOSITIONS 2:
(1) Competition for resources can lead to individuals with favorable traits reproducing more AND Natural selection is the process
      where organisms with favorable traits are more likely to reproduce
(2) Competition is a natural occurrence when resources are limited AND Unlimited resources can lead to an absence of competition
      AND Without competition, there's no natural selection
(3) The limited availability of a required resource can make survival and growth difficult for a species AND if an organism
      survives competition for resources then this organism will survive / reproduce for natural selection
(4) competition for resources is a selection pressure AND in nature, resources are limited AND natural selection occurs when there
       is competition for resources
(5) competition for unlimited resources is necessary for natural selection to occur AND limited availability of a required
      resource can make survival and growth difficult for a species
(6) natural selection requires competition for unlimited resources AND the survival of species depends on available resources and
      lack of competition
(7) competition for unlimited resources is a form of competition that drives natural selection AND natural selection is the
      process by which organisms with beneficial traits survive competition to reproduce and pass on those traits
(8) The survival of species depends on available resources and lack of competition AND limited availability of water can be a
      selection pressure leading to evolutionary adaptations AND without photosynthesis, a plant cannot produce the food it needs
      to grow and survive
(9) Competition for unlimited resources could be treated as a driving pressure for evolution AND Natural selection is one of the
      mechanisms of evolution
(10) competition for resources is a form of competition that drives natural selection AND natural selection is the process by
      which organisms with beneficial traits survive competition to reproduce and pass on those traits
(11) competition for unlimited resources provides a greater opportunity for organisms with beneficial traits to have an advantage
      in survival and reproduction AND natural selection requires competition for resources in order for organisms with beneficial
       traits to have an advantage in survival and reproduction
(12) The survival of species depends on available resources and lack of competition AND in nature, resources are limited

JUDGEMENTS 2 (12 items):
{{"index": 1, "factuality": [5, 5], "relevance": [5, 5], "redundancy": [5, 5], "complete_inference": 3, "explanation": "The
      decomposition does not specify the role of 'unlimited resources' in the competition."}}
{{"index": 2, "factuality": [5, 5, 5], "relevance": [5, 1, 5], "redundancy": [5, 5, 5], "complete_inference": 2, "explanation": "
      The second premise weakens the hypotheses, the first premise is irrelevant, and the last premise doesn't address unlimited
      resources"}}
{{"index": 3, "factuality": [5, 5], "relevance": [5, 5], "redundancy": [5, 5], "complete_inference": 2, "explanation": "The
      decomposition does not specify the role of 'unlimited resources' in the competition."}}
{{"index": 4, "factuality": [5, 5, 5], "relevance": [5, 2, 5], "redundancy": [5, 5, 5], "complete_inference": 2, "explanation": "
      The decomposition does not specify the role of 'unlimited resources' in the competition, and premise 2 is not very relevant
      "}}
{{"index": 5, "factuality": [5, 5], "relevance": [5, 5], "redundancy": [1, 5], "complete_inference": 2, "explanation": "the first
      premise restates the hypothesis"}}
{{"index": 6, "factuality": [5, 1], "relevance": [5, 5], "redundancy": [5, 5], "complete_inference": 2, "explanation": "The second
       premise is untrue, the survival of species does not depend on a lack of competition."}}
{{"index": 7, "factuality": [1, 5], "relevance": [5, 5], "redundancy": [5, 5], "complete_inference": 3, "explanation": "premise 1
      is false."}}
{{"index": 8, "factuality": [1, 5, 5], "relevance": [1, 1, 1], "redundancy": [5, 5, 5], "complete_inference": 1, "explanation": "
      The premises are about water and photosynthesis, not competition for unlimited resources."}}
{{"index": 9, "factuality": [2, 5], "relevance": [5, 5], "redundancy": [5, 5], "complete_inference": 2, "explanation": "premise 1
      is effectively untrue, and the decomposition does not explain how natural selection requires the competition for resources
      ."}}
{{"index": 10, "factuality": [5, 5], "relevance": [5, 5], "redundancy": [3, 5], "complete_inference": 4, "explanation": "Premise 1
       is somewhat redundant given premise 2, and the decomposition does not clearly explain why unlimited resources specifically
      are necessary for natural selection to occur."}}
{{"index": 11, "factuality": [1, 5], "relevance": [5, 5], "redundancy": [5, 5], "complete_inference": 3, "explanation": "premise 1
       is untrue."}}
{{"index": 12, "factuality": [1, 5], "relevance": [1, 1], "redundancy": [5, 5], "complete_inference": 2, "explanation": "The first
       premise is untrue, the survival of species does not depend on a lack of competition, and the second is irrelevant to the
      question of unlimited resources"}}

QUESTION 3:
The gravitational force between the Moon and Earth depends on (A) their masses, only, (B) their diameters, only, (C) their masses
      and how far apart they are, (D) their diameters and how far apart they are

HYPOTHESIS 3 (RECURSIVE):
the Moon and Earth are two objects

DECOMPOSITIONS 3:
(1) the Moon and Earth are at a certain distance apart AND the Moon and Earth have masses
(2) Earth is an object AND planets are objects AND the Moon is an object
(3) Earth is an object AND natural satellites are objects AND the Moon is a natural satellite
(4) the Moon and Earth have diameters AND the Moon and Earth have masses
(5) the Moon and Earth are at a certain distance apart AND the Moon and Earth have diameters
(6) all existing entities in space are regarded as objects AND the Earth exists AND the Moon exists
(7) the Earth is an object AND the Moon is an object AND two objects can exert gravitational force on each other
(8) Earth is an object AND the Moon is an object
(9) objects with mass are considered as objects in physics AND the Moon and Earth both have mass AND the Earth has mass
(10) the Earth is an object AND the Moon is an object
```

Figure 8: In-context learning prompt for using the RDTE protocol (2/3)

RDTE judgment prompt (continued):

```
JUDGEMENTS 3 (10 items):
{{"index": 1, "factuality": [5, 5], "relevance": [1, 5], "redundancy": [5, 5], "complete_inference": 2, "explanation": "premise 1
      is irrelevant to the hypothesis"}}
{{"index": 2, "factuality": [5, 5, 5], "relevance": [5, 5, 5], "redundancy": [5, 2, 5], "complete_inference": 4, "explanation": "
      that planets are objects is unnecessary given premise 1"}}
{{"index": 3, "factuality": [5, 5, 5], "relevance": [5, 5, 5], "redundancy": [5, 5, 5], "complete_inference": 5, "explanation": "
      The premises correctly entail that the Earth and the Moon (a natural satellite) are both objects."}}
{{"index": 4, "factuality": [5, 5], "relevance": [1, 5], "redundancy": [5, 5], "complete_inference": 2, "explanation": "premise 1
      is irrelevant"}}
{{"index": 5, "factuality": [5, 5], "relevance": [1, 1], "redundancy": [5, 5], "complete_inference": 1, "explanation": "both
      premises are irrelevant to the hypothesis"}}
{{"index": 6, "factuality": [5, 5, 5], "relevance": [5, 5, 5], "redundancy": [5, 5, 5], "complete_inference": 5, "explanation": "
      correctly identifies that the Earth and the Moon, which both exist, are considered objects in space."}}
{{"index": 7, "factuality": [5, 5, 5], "relevance": [5, 5, 1], "redundancy": [5, 5, 5], "complete_inference": 3, "explanation": "
      the third premise about gravitational force is not necessary to prove the hypothesis."}}
{{"index": 8, "factuality": [5, 5], "relevance": [5, 5], "redundancy": [5, 5], "complete_inference": 5, "explanation": "The
      premises properly entail that both things are objects."}}
{{"index": 9, "factuality": [5, 5, 5], "relevance": [5, 5, 5], "redundancy": [5, 5, 1], "complete_inference": 3, "explanation": "
      Premise 3 is redundant given premise 2"}}
{{"index": 10, "factuality": [5, 5], "relevance": [5, 5], "redundancy": [5, 5], "complete_inference": 5, "explanation": "directly
      states that both the Earth and the Moon are objects."}}

QUESTION 4:
{question}

HYPOTHESIS 4 {recursive_or_not}:
{hypothesis}

DECOMPOSITIONS 4:
{decompositions}

JUDGEMENTS 4 ({n_items} items):
```

Figure 9: In-context learning prompt for using the RDTE protocol (3/3)

Algorithm 1: Greedy End-to-End Entailment Tree Generator

Input : A list of queries $Q = [q_1, \ldots, q_n]$
Output : $\leq t$ scored trees for each query in Q

foreach q_i **in** Q **do**
  // Retrieve a set of support facts S conditioned on q_i
  $S \leftarrow \text{RetrieveSupportFacts}(q_i)$
  // Generate m candidate trees using ICL Prompt
  $T_1, \ldots, T_t \leftarrow \text{GenerateCandidateTrees}(q_i, S, m, t)$
  foreach T_j **in** T **do**
    // Prune disconnected nodes from the tree
    $T_j \leftarrow \text{PruneTree}(T_j)$
    // Check for ungrounded leaves not in S
    $U \leftarrow \text{FindUngroundedLeaves}(T_j, S)$
    foreach l_k **in** U **do**
      // Check if the leaf is entailed by a fact in the full index
      if not $\text{IsEntailedByIndex}(l_k)$ **then**
        continue
      end
    end
    // Retain tree if all leaves are grounded or entailed
    $\text{RetainTree}(T_j)$
  end
end
foreach retained tree T **do**
  // grade the tree using ChatGPT (student)
  $\text{ScoreTree}(T)$
end

```
In the following exercise, I would like you to tell me if a line of reasoning is reasonable or not.

I will give you some facts and a possible conclusion. Please tell me whether the conclusion reasonably follows from the facts I
      gave you. If the conclusion does reasonably follow from the facts, then please answer "yes". If the conclusion does not
      reasonably follow from the facts, then please answer "no".

Note that some of the facts may be false, but I am only interested whether the conclusion would reasonably follow IF those facts
      were true. In other words, imagine a world in which the given facts are true. Would it be reasonable to draw the conclusion
      from those facts, if they were true?

Here are some examples:

IF Vegetables are plants.
AND Cabbages are plants.
THEN Cabbages are vegetables.
Q: Does the rule's conclusion reasonably follow from the facts in the condition, if they were true? A: no

IF a nail is made of metal
AND metals conduct electricity
THEN a nail conducts electricity.
Q: Does the rule's conclusion reasonably follow from the facts in the condition, if they were true? A: yes

IF dogs are birds
AND birds can fly
THEN dogs can fly
Q: Does the rule's conclusion reasonably follow from the facts in the condition, if they were true? A: yes

IF sound requires matter to travel
AND a vacuum has no matter in it
THEN sound will not travel in a vacuum.
Q: Does the rule's conclusion reasonably follow from the facts in the condition, if they were true? A: yes

IF Erosion can cause a landslide.
AND Mud is deposited by a landslide.
THEN Erosion can cause mud to be deposited.
Q: Does the rule's conclusion reasonably follow from the facts in the condition, if they were true? A: yes

IF An animal needs to breathe in order to live.
AND Living things need water to live.
THEN Animals need water to live.
Q: Does the rule's conclusion reasonably follow from the facts in the condition, if they were true? A: yes

IF Frogs also have a larynx, or voice box, to make sounds.
AND Animals that have vocal cords can make sounds.
THEN Frogs are animals.
Q: Does the rule's conclusion reasonably follow from the facts in the condition, if they were true? A: no

IF All humans breathe.
AND Stones breathe.
THEN All humans and stones breathe.
Q: Does the rule's conclusion reasonably follow from the facts in the condition, if they were true? A: yes

IF If a planet is rocky, it can only have a thin atmosphere.
AND Small planets and rocky planets have very thin atmospheres.
THEN If a planet is small and rocky, it has a thin atmosphere.
Q: Does the rule's conclusion reasonably follow from the facts in the condition, if they were true? A: yes

IF Damming a river can cause a lake to form.
AND Dams are made of concrete.
THEN Dams are concrete lakes.
Q: Does the rule's conclusion reasonably follow from the facts in the condition, if they were true? A: no

Now your turn!
{Entailment}
```

```
Answer the following yes/no question with either "yes" or "no". Just give a single word answer. Do not give any explanation or
      justification.

Here are some examples:
Is it true that an ocean contains large bodies of water? yes
Is it true that lightning is similar to a volcano erupting? no
Is it true that a fox squirrel is a kind of animal? yes
Is it true that a rainbow is a kind of electromagnetic discharge? no
Is it true that the surface of the moon is made up of water? no
Is it true that the surface of the moon is made up of gases? no
Is it true that a bluebird is a kind of animal? yes
Is it true that the moon 's surface is made up of oceans? no
Is it true that the opposite of negative impact is positive impact? yes
Is it true that building a new highway through the area has a negative impact on the ecosystem? yes

Now let's do some more! Remember, answer with just a single word, yes or no.
Is it true that} \normalsize {\it insert the statement to assess here}
```

Figure 10: BaRDa prompts from Clark et al. (2023) for entailment (upper) and factuality (lower) judgments.

Document-conditioned forward generation prompt:

```
You are a reasoning system that searches for proofs of a hypothesis by decomposing into simpler premises.

For the following hypothesis and source documents, write a set of independent inferences entailed by one or multiple documents. The inferences should resemble world facts and should help to decompose the hypothesis into component reasoning steps. The inferences should NOT simply restate the hypothesis.

Your output should be a serialized json item, one per line, with the format {{"inference": <inference text>, "source": [<indices of source documents>]}} and nothing else.

QUESTION:
{question}

HYPOTHESIS:
{hypothesis}

SOURCE DOCUMENTS YOU MIGHT PULL FROM:
{documents}

{n} INFERENCES THAT MIGHT SUPPORT HYPOTHESIS:
```

Figure 11: Prompt used for creating forward-chaining inferences from retrieved source documents

Exemplar-conditioned decomposition generation prompt:

```
You are a reasoning system that searches for proofs of a hypothesis by decomposing into simpler premises.

Given a question and corresponding hypothesis, you give a list of 20 possible decompositions of the hypothesis into two or three facts, F1 and F2 (and possibly F3), such that proving the list of Fs would amount to proving the hypothesis through compositional entailment. There should be minimal "information loss" between the hypothesis and the Fs; you are looking for strict entailment.

Each decomposition should be some combination of core scientific principles as well as conclusions about the question at hand. They should not imply each other, i.e. none of them should start with "thus" or "therefore".

You also optionally take a list of facts that you might use in your decompositions.

Your 20 decompositions should follow different "reasoning patterns." Try to create decompositions that are semantically distinct and make use of different core facts or underlying principles.

Your output should be a serialized json item, one per line, with the format {"fact1": <fact1>, "fact2": <fact2>, "fact3" : <fact3, if necessary>} and nothing else.

QUESTION:
An ecosystem is a community of organisms interacting with their physical environment. Why are decomposers an important part of ecosystems? (A) They break down dead organisms to return nutrients to the soil. (B) They produce their own food for survival. (C) They play a role in preventing weathering and erosion. (D) They provide most of the energy to consumers.

HYPOTHESIS:
Decomposers are an important part of ecosystems because they break down dead organisms to return nutrients to the soil.

4 DIFFERENT POSSIBLE DECOMPOSITIONS, 2 OR 3 FACTS EACH, ONE JSON ITEM PER LINE:
{"fact1": "a decomposer breaks down dead organisms to return nutrients to soil", "fact2": "nutrients in soil are important for an ecosystem"}
{"fact1": "decomposition is when a decomposer breaks down dead organisms", "fact2": "decomposition is when a decomposer recycles / returns nutrients / nitrogen from dead organisms to the soil by eating those dead organisms", "fact3": "nutrients in soil are important for an ecosystem"}
{"fact1": "a decomposer breaks down dead organisms to return nutrients to soil", "fact2": "nutrients in soil are important to plants", "fact3": "plants are a part of an ecosystem"}

QUESTION:
What is the role of decomposers in a food chain? (A) They consume other organisms. (B) They break down dead organic matter. (C) They use the Sun's energy to make food. (D) They convert inorganic matter into organic matter.

HYPOTHESIS:
The role of decomposers in a food chain is they break down dead organic matter.

2 DIFFERENT POSSIBLE DECOMPOSITIONS, 2 OR 3 FACTS EACH, ONE JSON ITEM PER LINE:
{"fact1": "an organism is a source of organic matter", "fact2": "decomposer is a kind of role in the food chain process / in an ecosystem", "fact3": "decomposition is when a decomposer breaks down dead organisms"}
{"fact1": "an organism is a source of organic matter", "fact2": "the role of decomposers in the food chain process is to break down dead organisms"}

QUESTION:
{question}

HYPOTHESIS:
{hypothesis}

20 DIFFERENT POSSIBLE DECOMPOSITIONS, 2 OR 3 FACTS EACH, ONE JSON ITEM PER LINE::
```

Figure 12: Prompt used for creating exemplar-conditioned decompositions. Exemplars are retrieved from the EntailmentBank training set using BM25 with the question and hypothesis as query.

Fact-conditioned decomposition generation prompt:

```
You are a reasoning system that searches for proofs of a hypothesis by decomposing into simpler premises.

Given a question and corresponding hypothesis, you give a list of {n_candidates} possible decompositions of the hypothesis into
      two or three facts, F1 and F2 (and possibly F3), such that proving the list of Fs would amount to proving the hypothesis
      through compositional entailment. There should be minimal "information loss" between the hypothesis and the Fs; you are
      looking for strict entailment.

Each decomposition should be some combination of core scientific principles as well as conclusions about the question at hand.
      They should not imply each other, i.e. none of them should start with "thus" or "therefore".

You also optionally take a list of facts that you might use in your decompositions.

Your {n_candidates} decompositions should follow different "reasoning patterns." Try to create decompositions that are
      semantically distinct and make use of different core facts or underlying principles.

Your output should be a serialized json item, one per line, with the format {{"fact1": <fact1>, "fact2": <fact2>, "fact3" : <fact3
      , if necessary>}} and nothing else.

QUESTION:
{question}

HYPOTHESIS:
{hypothesis}

FACTS YOU MIGHT USE, IF THEY ARE RELEVANT:
{facts}

{n_candidates} DIFFERENT POSSIBLE DECOMPOSITIONS, ONE JSON ITEM PER LINE:

=============

textbf{(on followup)}
how could we make these better? regenerate the 20 decompositions so that they are higher-fidelity and are better explanations for
      the hypothesis.
```

Figure 13: Prompt used for generating fact-conditioned ad-hoc decompositions. The same prompt is re-used with an additional follow-up instruction to generate better decompositions.

Document-conditioned entailment prompt:

```
QUESTION:
{question}

HYPOTHESIS:
{hypothesis}

My student was trying to prove this hypothesis as it relates to the question. He pulled up this support document.

In the context of the QUESTION, does the PASSAGE entail the HYPOTHESIS? In other words, could we reasonably infer that the
      HYPOTHESIS is true in the context of the QUESTION using only the information in the PASSAGE?

PASSAGE:
{passage}

Please only make a judgment about whether the HYPOTHESIS is entailed by the PASSAGE, and not whether it answers the QUESTION.
Please score it on a scale of 1 to 5:

1: Definitely not entailed-- PASSAGE has nothing to do with the HYPOTHESIS
2: Poorly entailed-- PASSAGE might be on topic but does not provide any evidence for the HYPOTHESIS
3: Moderately entailed-- PASSAGE provides some evidence to suggest the HYPOTHESIS is true, but there is substantial missing
      information or ambiguity
4: Strongly entailed-- PASSAGE provides strong evidence for the HYPOTHESIS, but there is any amount of missing information or
      ambiguity
5: Definitely entailed-- PASSAGE provides strong evidence for the HYPOTHESIS, and there is no missing information or ambiguity
```

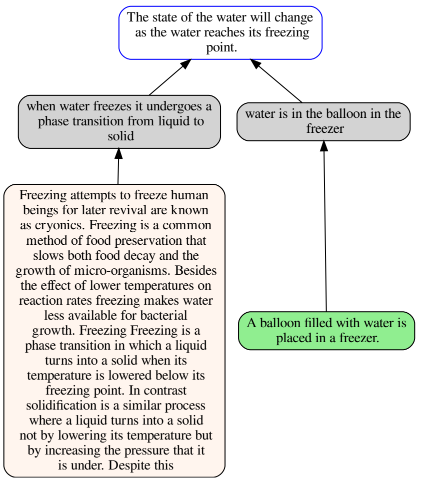
Figure 14: Prompt used to filter and score passage-hypothesis entailment pairs.

Algorithm 2: Stepwise Entailment Tree Generator

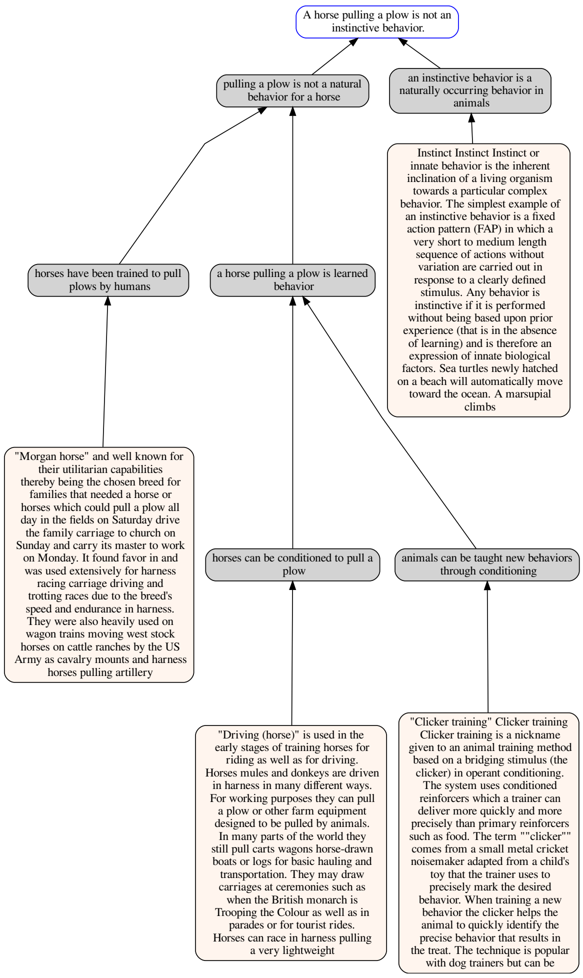
```
Input   : A list of queries Q = [q_1, ..., q_n]
Output  : t scored trees for each query in Q

foreach t in 1, 2, ... t do
    foreach q_i in Q do
        // Initialize search frontier and decompositions
        F ← {q_i}
        D ← []
        N ← 0
        while F ≠ ∅ and N < 10 do
            N ← N + 1
            // Retrieve and flatten support facts for sentences in F
            S ← Set(Flatten(RetrieveSupportFacts(f) for f in F))
            // Generate one line of tree decomposition using ICL prompt
            d_N ← GenOneStep(q_i, S, D)
            // Append line to decompositions
            D ← D + [d_N]
            // Create tree from decompositions
            T_N ← CreateTree(D)
            // Prune disconnected nodes from the tree
            T_N ← PruneTree(T_N)
            // Check for ungrounded leaves
            U ← FindUngroundedLeaves(T_N, S)
            foreach l_i in U do
                // Check if the leaf is entailed by a fact f in the full index
                if IsEntailedByIndex(l_i, f) then
                    // If it is, add the entailment to the decomposition list
                    D ← D + ["l_i ⇐ f"]
                    U ← U \ l_i
                end
            end
            // Set F to be the remaining ungrounded leaves
            F ← U
        end
    end
    Retain T_N
end
foreach retained tree T do
    // grade the tree using ChatGPT (student)
    ScoreTree(T)
end
```

A balloon filled with water is placed in a freezer. Which property of the water will change as the water reaches its freezing point? (A) color, (B) mass, **(C) state**, (D) weight



Which of these is not an instinctive behavior? (A) a bird building a nest, (B) a turtle burying its eggs, (C) a bear hibernating in winter, (D) a horse pulling a plow



A laser beam is aimed at four different objects. Through which of these objects will the laser beam pass and be refracted? (A) a black cloth, (B) a piece of aluminum, (C) a sheet of paper, **(D) a glass prism**

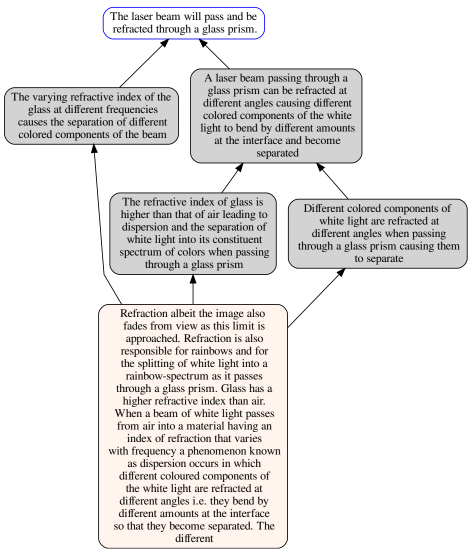
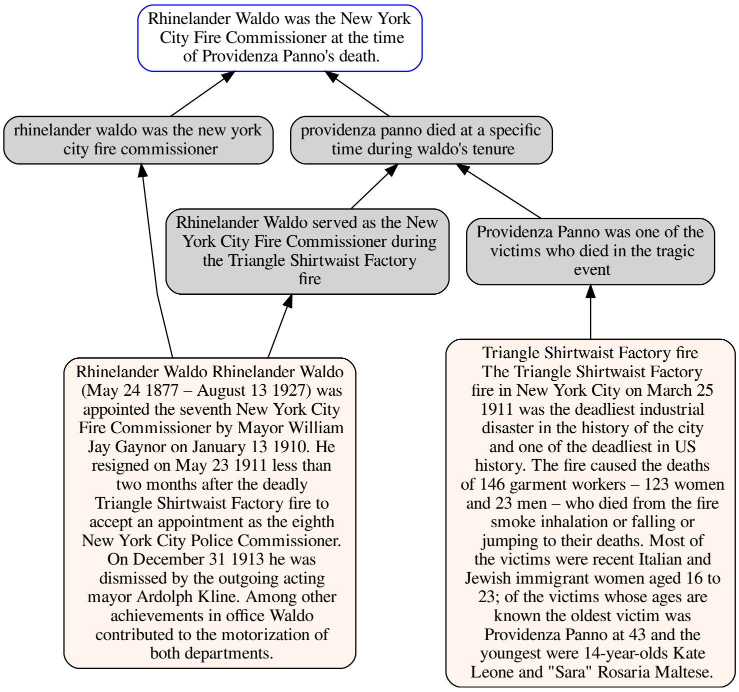


Figure 15: Example multiple-choice questions from ARC with TREEWISE's answer and corresponding proof grounded in Wikipedia.

Who was the New York City Fire Commissioner at the time of Providenza Panno's death? (A) Nicholas Scoppetta, (B) John J. Scannell, (C) William K. King, (D) Albert M. Arroyo, (E) Charles A. La Guardia, (F) Rhinelander Waldo, (G) James E. Langdon



Which Air Force member was behind enemy lines for 11 1/2 days and had the largest,longest and most complex rescue mission?

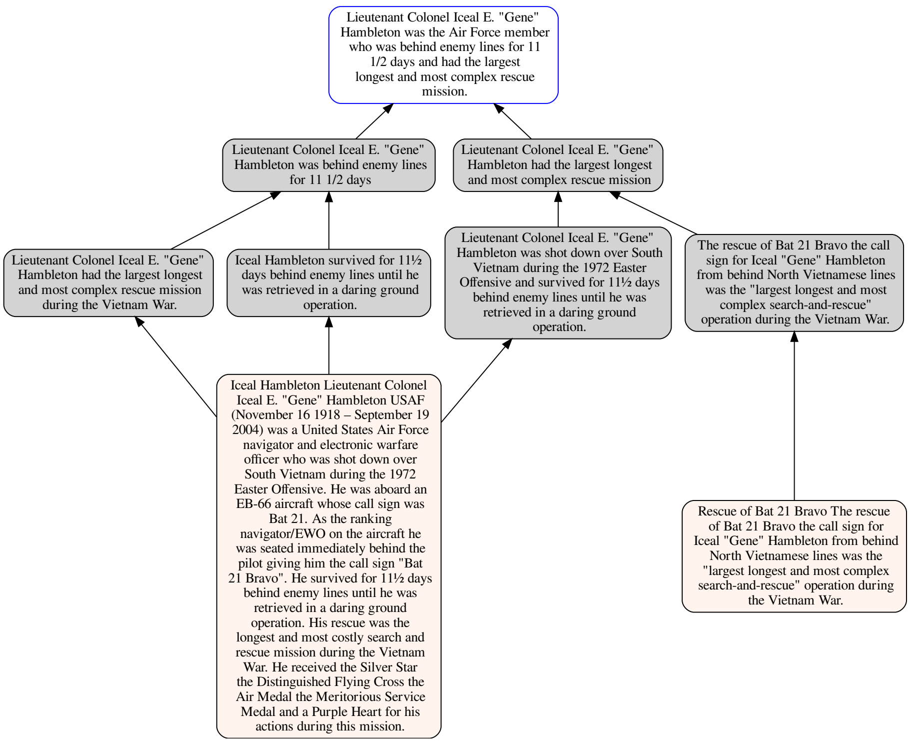


Figure 16: Example multiple-choice questions from HotpotQA with TREEWISE's answer and corresponding proof grounded in Wikipedia.