Explainable Inference Over Grounding-Abstract Chains for Science Questions

Mokanarangan Thayaparan[†], **Marco Valentino**[†], **André Freitas**[†‡]
Department of Computer Science, University of Manchester, United Kingdom[†]
Idiap Research Institute, Switzerland[‡]
{firstname.lastname}@manchester.ac.uk

Abstract

We propose an explainable inference approach for science questions by reasoning on grounding and abstract inference chains. This paper frames question answering as a *natural language* abductive reasoning problem, constructing *plausible* explanations for each candidate answer and then selecting the candidate with the *best* explanation as the final answer. Our method, *ExplanationLP*, elicits explanations by constructing a weighted graph of relevant facts for each candidate answer and employs a linear programming formalism designed to select the optimal subgraph of explanatory facts. The graphs' weighting function is composed of a set of parameters targeting relevance, cohesion and diversity, which we fine-tune for answer selection via Bayesian Optimisation. We carry out our experiments on the WorldTree and ARC-Challenge datasets to empirically demonstrate the following contributions: (1) ExplanationLP obtains strong performance when compared to transformer-based and multi-hop approaches despite having a significantly lower number of parameters; (2) We show that our model is able to generate plausible explanations for answer prediction; (3) Our model demonstrates better robustness towards semantic drift when compared to transformer-based and multi-hop approaches.

1 Introduction

Answering science questions remain a fundamental challenge in Natural Language Processing and AI as it requires complex forms of inference, including causal, model-based and example-based reasoning (Jansen, 2018; Clark et al., 2018; Jansen et al., 2016; Clark et al., 2013). Current state-of-the-art (SOTA) approaches for answering questions in the science domain are dominated by transformer-based models (Devlin et al., 2019; Sun et al., 2019). Despite remarkable performance on answer prediction, these approaches are black-box by nature, lacking the capability of providing *explanations* for their predictions (Thayaparan et al., 2020; Miller, 2019; Biran and Cotton, 2017; Jansen et al., 2016).

Explainable Science Question Answering (XSQA) is often framed as a *natural language abductive reasoning* problem (Khashabi et al., 2018; Jansen et al., 2017). Abductive reasoning represents a distinct inference process, known as *inference to the best explanation* (Peirce, 1960; Lipton, 2017), which starts from a set of complete or incomplete observations to find the hypothesis, from a set of plausible alternatives, that *best* explains the observations. Several approaches (Khashabi et al., 2018; Jansen et al., 2017; Khot et al., 2017a; Khashabi et al., 2016) employ this form of reasoning for multiple-choice science questions to build a set of plausible explanations for each candidate answer and select the one with the best explanation as the final answer.

XSQA solvers typically treat explanation generation as a multi-hop graph traversal problem. Here, the solver attempts to compose multiple facts that connect the question to a candidate answer. These *multi-hop* approaches have shown diminishing returns with an increasing number of hops (Jansen et al., 2018; Jansen, 2018). Fried et al. (2015) conclude that this phenomenon is due to *semantic drift* – i.e., as the number of aggregated facts increases, so does the probability of drifting out of context. Khashabi et al. (2019) propose a theoretical framework, empirically supported by Jansen et al. (2018); Fried et al. (2015), attesting that ongoing efforts with *very long* multi-hop reasoning chains are unlikely to succeed, emphasising the need for a *richer* representation with fewer hops and higher importance to abstraction and grounding mechanisms.

Consider the example in Figure 1A where the central concept the question examines is the understanding of *friction*. Here, an inference solver's

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 1–12

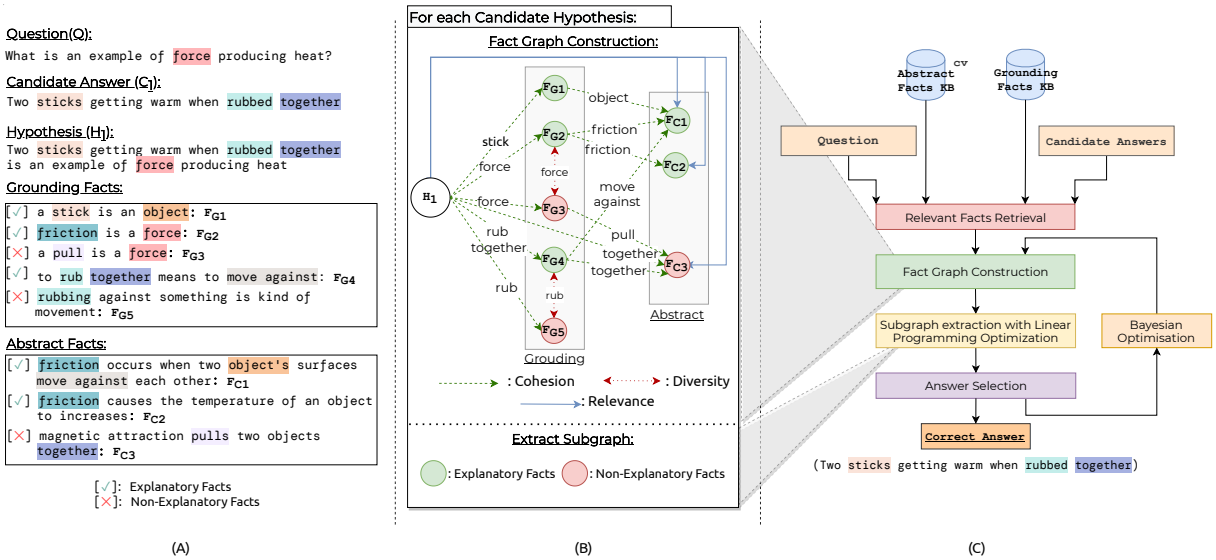

Figure 1: Overview of our approach: (A) Depicts a question, answer and formulated hypothesis along with the set of facts retrieved from a fact retrieval approach (B) Illustrates the optimisation process behind extracting explanatory facts for the provided hypothesis and facts. (C) Details the end-to-end architecture diagram.

challenge is to identify the core scientific facts (**Abstract Facts**) that best explain the answer. To achieve this goal, a QA solver should be able first to go from *force* to *friction*, *stick* to *object* and *rubbing together* to *move against*. These are the **Grounding Facts** that link generic or abstract concepts in a core scientific statement to specific terms occurring in question and candidate answer (Jansen et al., 2018). The grounding process is followed by the identification of the abstract facts about *friction*. A complete explanation for this question would require the composition of five facts to derive the correct answer successfully. However, it is possible to reduce the global reasoning in two hops, modelling it with grounding and abstract facts.

In line with these observations, this work presents a novel approach that explicitly models abstract and grounding mechanisms. The contributions of the paper are:

1. We present a novel approach that performs natural language abductive reasoning via grounding-abstract chains combining Linear Programming with Bayesian optimisation for science question answering (Section 2).

2. We obtain comparable performance when compared to transformers, multi-hop approaches and previous Linear Programming models despite having a significantly lower number of parameters (Section 3.1).

3. We demonstrate that our model can generate plausible explanations for answer prediction (Section 3.2) and validate the importance of grounding-abstract chains via ablation analysis (Section 3.3).

2 ExplanationLP: Abductive Reasoning with Linear Programming

ExplanationLP answers and explains multiple-choice science questions via abductive natural language reasoning. Specifically, the task of answering multiple-choice science questions is reformulated as the problem of finding the candidate answer that is supported by the best explanation. For each Question Q and candidate answer $c_i \in C$, ExplanationLP converts to a hypothesis h_i and attempts to construct a plausible explanation.

Figure 1C illustrates the end-to-end framework. From an initial set of facts selected using a retrieval model, ExplanationLP constructs a fact graph where each node is a fact, and the nodes and edges have a score according to three properties: *relevance*, *cohesion* and *diversity*. Subsequently, an optimal subgraph is extracted using Linear Programming, whose role is to select the best sub-set of facts while preserving structural constraints imposed via grounding-abstract chains. The subgraphs' global scores computed by summing up the nodes and edges scores are adopted to select the final answer. Since the subgraph scores depend on the sum of nodes and edge scores, each property is multiplied by a learnable weight which

is optimised via Bayesian Optimisation to obtain the best possible combination with the highest accuracy for answer selection. To the best of our knowledge, we are the first to combine a parameter optimisation method with Linear Programming for inference. The rest of this section describes the model in detail.

2.1 Relevant facts retrival

Given a question (Q) and candidate answers $C = \{c_1,\ c_2,\ c_3, ...,\ c_n\}$ we convert them to hypotheses $\{h_1,\ h_2,\ h_3, ...,\ h_n\}$ using the approach proposed by Demszky et al. (2018). For each hypothesis h_i we adopt fact retrieval approaches (e.g: BM25, Unification-retrieval (Valentino et al., 2021)) to select the top m relevant *abstract* facts $F_A^{h_i} = \{f_1^{h_i}, f_2^{h_i}, f_3^{h_i}, ..., f_m^{h_i}\}$ from a knowledge base containing abstract facts (*Abstract Facts KB*) and top l relevant *grounding* facts $F_G^{h_i} = \{f_1^{h_i}, f_2^{h_i}, f_3^{h_i}, ..., f_l^{h_i}\}$ from a knowledge base containing grounding facts (*Grounding Facts KB*) that *at least* connects one abstract fact with the hypothesis, such that $F^{h_i} = F_A^{h_i} \cup F_G^{h_i}$ and $l+m = k$.

2.2 Fact graph construction

For each hypothesis h_i we build a weighted undirected graph $G^{h_i} = (V^{h_i},\ E^{h_i},\ \omega_v,\ \omega_e)$ with vertices $V^{h_i} \in \{\{h_i\} \cup F^{h_i}\}$, edges E^{h_i}, edge-weight function $\omega_e(e_i; \theta_1)$ and node-weight function $\omega_v(v_i; \theta_2)$ where $e_i \in E^{h_i}$, $v_i \in V^{h_i}$ and $\theta_1, \theta_2 \in [0, 1]$ is a learnable parameter which is optimised via Bayesian optimisation.

The model scores the nodes and edges based on the following *three* properties (See Figure 1B):

(1) Relevance: We promote the inclusion of highly relevant facts in the explanations by encouraging the selection of sentences with higher lexical relevance and semantic similarity with the hypothesis. We use the following scores to measure the relevance and the semantic similarity of the facts:

Lexical Relevance score (L): Obtained from the upstream facts retrieval model (e.g: BM25 score/ Unification score (Valentino et al., 2021)).

Semantic Similarity score (S): Cosine similarity obtained from neural sentence representation models. For our experiments, we adopt Sentence-BERT (Reimers et al., 2019) since it shows state-of-the-art performance in semantic textual similarity tasks.

(2) Cohesion: Explanations should be cohesive, implying that grounding-abstract chains should remain within the same context. To achieve cohesion, we encourage a high degree of overlaps between different hops (e.g. hypothesis-grounding, grounding-abstract, hypothesis-abstract) to prevent the inference chains from drifting away from the original context. The overlap across two hops is quantified using the following scoring function:

Cohesion score (C): We denote the set of unique terms of a given fact $f_i^{h_i}$ as $t(f_i^{h_i})$ after being lemmatized and stripped of stopwords. The overlap score of two facts $f_j^{h_i}$ and $f_j^{h_i}$ is given by:

$$C(f_j^{h_i}, f_k^{h_i}) = \frac{|t(f_j^{h_i}) \cap t(f_k^{h_i})|}{max(|t(f_j^{h_i})|, |t(f_k^{h_i})|)}$$

Therefore, the higher the number of term overlaps, the higher the cohesion score.

(3) Diversity: While maximizing relevance and cohesion between different hops, we encourage diversity between facts of the same type (e.g. abstract-abstract, grounding-grounding) to address different parts of the hypothesis and promote completeness in the explanations. We measure diversity via the following function:

Diversity score (D): We denote the overlaps between hypothesis h_i and the fact $f_i^{h_i}$ as $t_{h_i}(f_i^{h_i}) = t(f_i^{h_i}) \cap t(h_i)$. The diversity score of two facts $f_j^{h_i}$ and $f_j^{h_i}$ is given by:

$$D(f_j^{h_i}, f_k^{h_i}) = -1\frac{|t_{h_i}(f_j^{h_i}) \cap t_{h_i}(f_k^{h_i})|}{max(|t_{h_i}(f_j^{h_i})|, |t_{h_i}(f_k^{h_i})|)}$$

The goal is to maximise diversity and avoid redundant facts in the explanations. Therefore, if two facts overlap with different parts of the hypothesis, they will have a higher diversity score compared to two facts that overlap with the same part.

Given these premises, the weight functions of the graph is designed as follows:

$$\omega_e(v_j, v_k; \theta_1) = \begin{cases} \theta_{gg} D(v_j, v_k) & v_j, v_k \in F_G^{h_i} \\ \theta_{aa} D(v_j, v_k) & v_j, v_k \in F_A^{h_i} \\ \theta_{ga} C(v_j, v_k) & v_j \in F_G^{h_i}, v_k \in F_A^{h_i} \\ \theta_{qg} C(v_j, v_k) & v_j \in F_G^{h_i}, v_k = h_i \\ \theta_{qa} C(v_j, v_k) & v_j \in F_A^{h_i}, v_k = h_i \end{cases}$$

$$\omega_v(v_i^{h_i}; \theta_2) = \begin{cases} \theta_{lr} L(v_j, h_i) + \theta_{ss} S(v_j, h_i) & v_j \in F_A^{h_i} \\ 0 & v_i \in F_G^{h_i} \\ 0 & v_i = h_i \end{cases}$$

where $\theta_{gg},\ \theta_{aa},\ \theta_{ga},\ \theta_{gq},\ \theta_{qa} \in \theta_1$ and $\theta_{lr},\ \theta_{ss} \in \theta_2$.

2.3 Subgraph extraction with Linear Programming (LP) optimisation

The construction of the explanation graph has to be optimised for the downstream answer selection task. Specifically, from the whole set of facts retrieved by the upstream retrieval models, we need to select the optimal subgraph that maximises the performance of answer prediction. To achieve this goal, we adopt a Linear Programming approach.

The selection of the explanation graph is framed as a rooted maximum-weight connected subgraph problem with a maximum number of K vertices (R-MWCS$_K$). This formalism is derived from the generalized maximum-weight connected subgraph problem (Loboda et al., 2016). R-MWCS$_K$ has two parts: objective function to be maximized and constraints to build a connected subgraph of explanatory facts. The formal definition of the objective function is as follows:

Definition 1. Given a connected undirected graph $G = (V, E)$ with edge-weight function $\omega_e : E \rightarrow \mathbb{R}$, node-weight function $\omega_v : V \rightarrow \mathbb{R}$, root vertex $r \in V$ and expected number of vertices K, the rooted maximum-weight connected subgraph problem with K number of vertices (R-MWCS$_K$) problem is finding the connected subgraph $\hat{G} = (\hat{V}, \hat{E})$ such that $r \in \hat{V}$, $|V| \leq K$ and

$$\Omega(\hat{G}; \theta_3) = \theta_{vw} \sum_{v \in \hat{V}} \omega_v(v; \theta_1) + \theta_{ew} \sum_{e \in \hat{E}} \omega_e(e; \theta_2) \rightarrow max$$

where $\theta_{vw},\ \theta_{ew} \in \theta_3, \theta_3 \in [0, 1]$ and θ_3 is a learnable parameter optimized via Bayesian optimisation. The LP solver will seek to extract the optimal subgraph with the highest possible sum of node and edge weights. Since the solver seeks to obtain the highest possible score, it will avoid negative edges and will prioritise high-value positive edges resulting in higher diversity, cohesion and relevance. We adopt the following binary variables to represent the presence of nodes and edges in the subgraph:

1. Binary variable y_v takes the value of 1 iff $v \in V^{h_i}$ belongs to the subgraph.

2. Binary variable z_e takes the value of 1 iff $e \in E^{h_i}$ belongs to the subgraph.

In order to emulate the grounding-abstract inference chains and obtain a valid subgraph, we impose the set constraints described in Table 1 for the LP solver.

2.4 Bayesian Optimisation for Answer Selection

Given Question Q and choices $C = \{c_1,\ c_2,\ c_3, ...,\ c_n\}$ we extract the optimal explanation graphs $\hat{G}^Q = \{\hat{G}^{c_1},\ \hat{G}^{c_2},\ \hat{G}^{c_3}, ...,\ \hat{G}^{c_n}\}$ for each choice. We consider the hypothesis with the highest relevance, cohesion and diversity to be the correct the answer. Based on this premise we define the correct answer as $c_{ans} = \arg\max_{h_i}\ (\Omega(\hat{G}^{h_i}))$.

In order to automatically optimize the Linear Programming model (i.e, $\theta_1,\ \theta_2,\ \theta_3$) we use Bayesian optimisation. The algorithm is defined as below (Here $\mathcal{GP}$ is Gaussian Process and LP is the Linear Programming module).

Algorithm 1: Bayesian Optimisation

$\theta_1, \theta_2, \theta_3$ = initRandom(seed)
G^Q = fact-graph-construction($\omega_e(\theta_1^{'}),\ \omega_v(\theta_2^{'})$)
$\hat{G}^Q$ = LP($G^Q,\ \Omega(\theta_3)$)
X = evaluate-accuracy(G^Q)
model = $\mathcal{GP}(X, \{\theta_1, \theta_2, \theta_3\})$
iteration = 0
while *iteration* $\leq N$ **do**
  $\theta_1^{'}, \theta_2^{'}, \theta_3^{'}$ = get-next-exploration-point()
  $G^{Q^{'}}$ = fact-graph-construction($\omega_e(\theta_1^{'}),\ \omega_v(\theta_2^{'})$)
  $\hat{G}^{Q^{'}}$ = LP($G^{Q^{'}},\ \Omega(\theta_3^{'})$)
  $X^{'}$ = evaluate-accuracy($G^{Q^{'}}$)
  model.update($X^{'}, \{\theta_1^{'}, \theta_2^{'}, \theta_3^{'}\}$)
  iteration = iteration + 1
end
Result: Best accuracy for model and respective parameters $\theta_1, \theta_2, \theta_3$

3 Empirical Evaluation

Background Knowledge: We construct the required knowledge bases using the following sources.

(1) Abstract KB: Our Abstract knowledge base is constructed from the WorldTree Tablestore corpus (Xie et al., 2020; Jansen et al., 2018). The Tablestore corpus contains a set of common sense and scientific facts adopted to create explanations for multiple-choice science questions. The corpus is built for answering elementary science questions encouraging possible knowledge reuse to elicit explanatory patterns. We extract the core scientific facts to build the Abstract KB. Core scientific facts

$y_{v_i} = 1 \quad \text{if } v_i = h_i \quad (1)$ $y_{v_i} \leq \sum_j y_{v_j} \quad \forall v_j \in N_{G^{h_i}}(v_i) \quad (2)$ $z_{v_i,v_j} \leq y_{v_i} \quad \forall e_{(v_i,v_j)} \in E \quad (3)$ $z_{v_i,v_j} \leq y_{v_j} \quad \forall e_{(v_i,v_j)} \in E \quad (4)$ $z_{v_i,v_j} \geq y_{v_i} + y_{v_j} - 1 \quad \forall e_{(v_i,v_j)} \in E \quad (5)$	**Chaining constraint**: Equation 1 states that the subgraph should always contain the hypothesis node. Inequality 2 states that if a vertex is to be part of the subgraph, then at least one of its neighbors with a lexical overlap should also be part of the subgraph. Equation 1 and Inequality 2 restrict the LP method to construct explanations that originate from the hypothesis and perform multi-hop aggregation based on the existence of lexical overlap. Inequalities 3, 4 and 5 state that if two vertices are in the subgraph then the edges connecting the vertices should be also in the subgraph. These inequality constraints will force the LP method to avoid grounding nodes with high overlap regardless of their relevance.
$\sum_i y_{v_i} \leq K \quad \forall v_i \in F_A^{h_i} \quad (6)$	**Abstract fact limit constraint**: Equation 6 limits the total number of abstract facts to K. Instead of limiting of total selected number of nodes to K, by limiting the abstract facts we dictate the need for grounding facts based on the number of terms present in the hypothesis and in the abstract facts.
$\sum_{v_j} y_{v_i} - 2 \geq -2(1 - y_{v_j}) \quad \forall v_i \in N_{G^{h_i}}(v_j), v_i \in \{F_A^{h_i} \cup h_i\}, v_j \in F_G^{h_i} \quad (7)$	**Grounding neighbor constraint**: Inequality 7 states that if a grounding fact is selected, then at least two of its neighbors should be either both abstract facts or a hypothesis and an abstract fact. This constraint ensures that grounding facts play the linking role connecting hypothesis-abstract facts.

Table 1: Linear programming constraints employed by ExplanationLP to emulate grounding-abstract inference chains and extract the optimal subgraph

are independent from the specific questions and represent general scientific and commonsense knowledge, such as `Actions` (*friction occurs when two object's surfaces move against each other*) or `Affordances` (*friction causes the temperature of an object to increase*).

(2) Grounding KB: The grounding knowledge base consists of definitional knowledge (e.g., synonymy and taxonomy) that can take into account lexical variability of questions and help it link it to abstract facts. To achieve this goal, we select the *is-a* and *synonymy* facts from ConceptNet (Speer et al., 2017) as our grounding facts. ConceptNet has high coverage and precision, enabling us to answer a wide variety of questions.

Question Sets: We use the following question sets to evaluate ExplanationLP's performance and compare it against other explainable approaches:

(1) WorldTree Corpus: The 2,290 questions in the WorldTree corpus are split into three different subsets: *train-set* (987), *dev-set* (226) and *test-set* (1,077). We use the *dev-set* to assess the explainability performance and robustness analysis since the explanations for *test-set* are not publicly available.

(2) ARC-Challenge Corpus: ARC-Challenge is a multiple-choice question dataset which consists of question from science exams from grade 3 to grade 9 (Clark et al., 2018). We only consider the Challenge set of questions. These questions have proven to be challenging to answer for other LP-based question answering and neural approaches. ExplanationLP rely only on the *train-set* (1,119) and test on the *test-set* (1,172). ExplanationLP does not require *dev-set*, since the possibility of over-fitting is non-existent with only ten parameters.

Relevant Facts Retrieval (FR): We experiment with two different fact retrieval scores. The first model – i.e. *BM25 Retrieval*, adopts a BM25 vector representation for hypothesis and explanation facts. We apply this retrieval for both Grounding and Abstract retrieval. We use the IDF score from BM25 as our downstream model's relevance score. The second approach – i.e. *Unification Retrieval (UR)*, represents the BM25 implementation of the Unification-based Reconstruction framework described in Valentino et al. (2021). The unification score for a given fact depends on how often the same fact appears in explanations for similar questions.

Baselines: The following baselines are replicated

on the WorldTree corpus to compare against ExplanationLP:

(1) Bert-Based models: We compare the ExplanationLP model's performance against a set of BERT baselines. The first baseline – i.e. *BERT*$_{Base}$/*BERT*$_{Large}$, is represented by a standard BERT language model (Devlin et al., 2019) fine-tuned for multiple-choice question answering. Specifically, the model is trained for binary classification on each question-candidate answer pair to maximize the correct choice (i.e., predict 1) and minimize the wrong choices (i.e., predict 0). During inference, we select the choice with the highest prediction score as the correct answer. BERT baselines are further enhanced with explanatory facts retrieved by the retrieval models. *BERT + BM25* and *BERT + UR*, is fine-tuned for binary classification by complementing the question-answer pair with grounding and abstract facts selected by BM25 and Unification retrieval, respectively.

Similarly, the second model *BERT + UR* complements the question-answer pair with grounding and abstract facts selected using BM25 and Unification retrieval, respectively.

(2) PathNet (Kundu et al., 2019): PathNet is a neural approach that constructs a single linear path composed of two facts connected via entity pairs for reasoning. PathNet also can explain its reasoning via explicit reasoning paths. They have exhibited strong performance for multiple-choice science questions by composing two facts. Similar to Bert-based models, we employ PathNET with the top k facts retrieved utilizing Unification (*PathNet + UR*) and BM25 (*PathNet + BM25*) retrieval. We concatenate the facts retrieved for each candidate answer and provide as supporting facts.

Further details regarding the hyperparameters and code used for each model, along with information concerning the knowledge base construction and dataset information, can be found in the Supplementary Materials.

3.1 Answer Selection

WorldTree Corpus: We retrieve the top l relevant grounding facts from Grounding KB and the top m relevant abstract facts from Abstract KB such that $l + m = k$ and $l = m$. To ensure fairness across the approaches, the same amount of facts are presented to each model. We experimented with $k = \{10,\ 20,\ 30,\ 40,\ 50\}$ and report the accuracy across Easy and Challenge split of the best performing setting in Table 2. We draw the following conclusions:

#	Model	Accuracy Easy	Accuracy Challenge
1	BERT$_{Base}$	51.04	28.75
2	BERT$_{Large}$	54.58	29.39
3	BERT$_{Base}$ + BM25 (*k*=10)	53.92	42.72
4	BERT$_{Large}$ + BM25 (*k*=10)	54.05	43.45
5	BERT$_{Base}$ + UR (*k*=10)	52.87	42.17
6	BERT$_{Large}$ + UR (*k*=10)	58.50	43.72
7	PathNet + BM25 (*k*=20)	43.32	36.42
8	PathNet + UR (*k*=15)	47.64	33.55
9	Ours + BM25 (*k*=30)	63.82	48.24
10	Ours + UR (*k*=30)	**66.23**	**50.15**

Table 2: Accuracy on Easy (764) and Challenge split (313) of WorldTree *test-set* corpus from the best performing *k* of each model

#	Model	Explainable	Accuracy
1	BERT$_{Large}$	No	35.11
2	IR Solver (Clark et al., 2016)	Yes	20.26
3	TupleInf (Khot et al., 2017b)	Yes	23.83
4	TableILP (Khashabi et al., 2016)	Yes	26.97
5	DGEM (Clark et al., 2016)	Partial	27.11
6	KG^2 (Zhang et al., 2018)	Partial	31.70
7	ET-RR (Ni et al., 2019)	Partial	36.61
8	Unsupervised AHE (Yadav et al., 2019a)	Partial	33.87
9	Supervised AHE (Yadav et al., 2019a)	Partial	34.47
10	AutoRocc (Yadav et al., 2019b)	Partial	**41.24**
11	Ours + BM25 (*k*=40)	Yes	**40.21**
12	Ours + UR (*k*=40)	Yes	39.84

Table 3: ARC challenge scores compared with other Fully or Partially explainable approaches trained *only* on the ARC dataset.

(1) Despite having a smaller number of parameters to train (BERT$_{Base}$: 110M parameters, BERT$_{Large}$: 340M parameters, ExplanationLP: 9 parameters), the best performing ExplanationLP (#10) overall outperforms all the BERT$_{Base}$ and BERT$_{Large}$ models on both Challenge and Easy split. We outperform the best performing BERT model with facts (BERT$_{Large}$ (#6)) by 7.74% in Easy and 6.43% in Challenge. We also outperform best performing BERT without facts (BERT$_{Large}$ (#2)) by 11.66% in Easy and 20.76% in Challenge.

(2) BERT is inherently a black-box model, not being entirely possible to explain its prediction. By contrast, ExplanationLP is fully explainable and produces a complete explanatory graph.

(3) Similar to ExplanationLP, PathNet is also explainable and demonstrates robustness to noise.

CASE I: All the selected facts are in the gold explanation (**Frequency**: 33%)
Question: A company wants to make a game that uses a magnet that sticks to a board. Which material should it use for the board? **Answer:** steel **Explanations:** (1) steel is a metal (*Grounding*), (2) if a magnet is attracted to a metal then that magnet will stick to that metal (*Abstract*), (3) a magnet attracts magnetic metals through magnetism (*Abstract*),
CASE II: At least one selected facts are in the gold explanation (**Frequency**: 58%)
Question: A large piece of ice is placed on the sidewalk on a warm day. What will happen to the ice? **Answer**: It will melt to form liquid water. **Explanations:** (1) drop is liquid small amount (*Grounding*), (2) forming something is change (*Grounding*), (3) ice wedging is mechanical weathering (*Grounding*), (4) melting means changing from a solid into a liquid by adding heat energy (*Abstract*), (5) weathering means breaking down surface materials from larger whole into smaller pieces by weather (*Abstract*),
CASE III: No retrieved facts is in the gold explanation (**Frequency**: 9%)
Question:Wind is a natural resource that benefits the southeastern shore of the Chesapeake Bay. How could these winds best benefit humans? **Answer:** The winds could be converted to electrical energy **Explanations:** (1) renewable resource is natural resource (*Grounding*), (2) wind is a renewable resource (*Abstract*), (3) electrical devices convert electricity into other forms of energy (*Abstract*)

Table 4: Case study of explanation extracted by ExplanationLP

ExplanationLP also outperforms PathNet's best performance setting (#8) by 18.59% in Easy and 16.60% in Challenge.

(4) ExplanationLP consistently exhibits better scores on both BM25 and UR than BERT and PathNet, demonstrating independence of the upstream retrieval model for performance.

ARC-Challenge : We also evaluated our model on the ARC-Challenge corpus (Clark et al., 2018) to evaluate ExplanationLP on a more extensive general question set and compare against contemporary approaches that provide explanations for an inference that has *only* been trained on ARC corpus. Table 3 reports the results on the *test-set*. We compare ExplanationLP against published approaches that are fully/partly explainable. Here explainability indicates if the model produces an explanation/evidence for the predicted answer. A subset of the approaches produces evidence for the answer but remains intrinsically black-box. These models have been marked as *Partial*.

As depicted in the Table 3, we outperform the best performing fully explainable (#4 TableILP) model by 13.28%. We also outperform specific neural approaches with larger parameter sets (#5 - #9) that provide explanations for their inference and BERT (#1). Despite having a smaller number of training parameters, we also exhibit competitive performance with a state-of-the-art Bert-based approach (#10) that do not use external resources to train the QA system.

3.2 Explainability

Approach	Precision	Recall	F1
PathNet + UR (k=20)	21.56	36.55	29.06
Ours + UR (k=30)	**57.96**	**49.92**	**48.13**

Table 5: Explanation retrieval performance on the WorldTree Corpus *dev-set*.

Table 5 shows the Precision, Recall and F1_{Macro} score for explanation retrieval for PathNet and ExplanationLP. These scores are computed using gold abstract explanations from WorldTree corpus. We outperform PathNet across all spectrum by a significant margin.

Table 4 reports three representative cases that show how explanation generation relates to correct answer prediction. The first example (Case I) represents the situation in which all the selected sentences are annotated as gold explanations in the WorldTree corpus (dev-set). The second example (Case II) shows the case in which at least one sentence in the explanation is labelled as gold. Finally, the third example (Case III) represents the case in which the explanation generated by the method does not contain any gold fact. We observe Case I and Case II occur over 91% of the questions, demonstrating that the correct answers are mostly derived from plausible explanations.

3.3 Ablation Study

In order to understand the contribution lent by different components, we choose the best setting

#	Approach	Accuracy WT	ARC
1	ExplanationLP **(Best)**	**61.37**	**40.21**
Structure			
2	Grounding-Abstract Categories	58.33	35.13
3	Edge weights	43.78	29.45
4	Node weights	42.80	27.87
Cohesion			
5	Hypothesis-Abstract cohesion	38.71	30.37
6	Hypothesis-Grounding cohesion	59.33	38.73
7	Grounding-Abstract cohesion	59.12	38.14
Diversity			
8	Abstract-Abstract diversity	60.16	37.62
9	Grounding-Grounding diversity	60.44	37.71
Relevance			
10	Hypothesis-Abstract semantic similarity	55.38	35.49
11	Hypothesis-Abstract lexical relevance	54.68	36.01

Table 6: Ablation study, removing different components of ExplanationLP. The scores reported here are accuracy for answer selection on the WorldTree (WT) and ARC-Challenge (ARC) test-set.

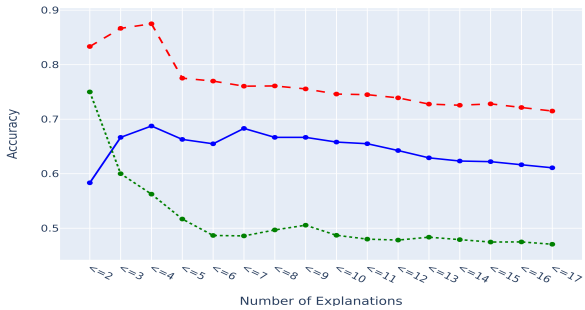

Figure 2: Change in accuracy of answer prediction the development set varying across different models with increasing explanation length for WorldTree *dev-set*. Red dashed line represents ExplanationLP + UR (k=30), blue line represents BERT$_{Large}$ + UR (k=10) and green dotted line represents PathNet + UR (k=20)

(*WorldTree*: ExplanationLP + UR (k=30) and *ARC*: ExplanationLP + BM25 (k=40)) and drop different components to perform an ablation analysis. We retain the ensemble after removing each component. The results are summarized in Table 6.

(1) The grounding-abstract chains (#2) play a significant role, particularly in the reasoning mechanism on a challenging question set like ARC-Challenge.

(2) As observed in #3, #4 removing node weights and edge weights lead to a dramatic drop in performance. This drop indicates that both are fundamental for the final prediction, highlighting the role of graph structure in explainable inference.

(3) The importance of cohesion varies across different types of facts. We observe that Hypothesis-Abstract cohesion (#5) is significantly more important than the others. We attribute this to the fact that without Hypothesis-Abstract cohesion, multi-hop inference can quickly go out of context.

(4) From the ablation analysis, we can see how lexical relevance and semantic similarity (#10, 11) complements each other towards the final prediction. For WorldTree corpus, the relevance score has a higher parameter score translating into a higher impact and vice-versa for ARC.

(5) Diversity plays a smaller role when compared to cohesion and relevance. The impact of diversity in ARC is higher than that of WorldTree.

Semantic Drift To validate the performance across an increasing number of hops, we plot the accuracy against explanation length as illustrated in Figure 2. As demonstrated in explanation regeneration (Valentino et al., 2021; Jansen and Ustalov, 2019), the complexity of a science question is directly correlated with the explanation length – i.e. the number of facts required in the gold explanation. Unlike BERT, PathNet and ExplanationLP use external background knowledge, addressing the multi-hop process in two main reasoning steps. However, in contrast to ExplanationLP, PathNet combines only two explanatory facts to answer a given question. This assumption has a negative impact on answering complex questions requiring long explanations. This is evident in the graph, where we observe a sharp decrease in accuracy with increasing explanation length. Comparatively, ExplanationLP achieves more stable performance, showing a lower degradation with an increasing number of explanation sentences. These results crucially demonstrate the positive impact of grounding-abstract mechanisms on semantic drift. We also exhibit consistently better performance when compared with BERT as well.

4 Related Work

Our approach broadly falls into Linear Programming based approaches for science question answering. LP-based approaches perform inference over either semi-structured tables (Khashabi et al., 2016) or structural representations extracted from the text (Khashabi et al., 2018; Khot et al., 2017a). These approaches treat all facts homogeneously and attempt to connect the question with the correct answer through long hops. While they have exhibited good performance with no supervision, the performance tends to be lower when answer-

ing complex questions requiring long explanatory chains. In contrast, our approach performs inference over unstructured text by imposing structural constraints via grounding-abstract chains, lowering the hops, and also combine parametric optimisation to extract the best performing model.

The other class of approaches that provide explanations are graph-based approaches. Graph-based approaches have been successfully applied for open-domain question answering (Fang et al., 2020; Qiu et al., 2019; Thayaparan et al., 2019) where the question only requires only two hops. PathNet (Kundu et al., 2019) operates within the same design principles and has been applied on OpenbookQA science dataset. As indicated in the empirical evaluation, it struggles with long-chain explanations since it relies only on two facts. Graph-based approaches have also been employed for mathematical reasoning (Ferreira and Freitas, 2020a,b) and textual entailment (Silva et al., 2019, 2018).

The third category of partially explainable approaches employs black-box neural models in combination with a retrieval approach. The SOTA model for Science Question (Khashabi et al., 2020) answering is pretrained across multiple datasets and is not explainable. The current partially explainable SOTA approach that does not rely on external resource (Yadav et al., 2019b) employs a large parameter BERT model for question answering resulting. In contrast, with a low number of parameters, we have introduced a model that demonstrates competitive performance and leaves a smaller carbon footprint in terms of energy consumption (Henderson et al., 2020). Other methods construct explanation chains by leveraging explanatory patterns emerging in a corpus of scientific explanations (Valentino et al., 2020, 2021).

5 Conclusion

This paper presented a robust, explainable and efficient science question answering model that performs abductive natural language inference. We also presented an in-depth systematic evaluation demonstrating the impact on the various set of design principles via an in-depth ablation analysis. Despite having a significantly lower number of parameters, we demonstrated competitive performance compared with contemporary explainable approaches while also showcasing its robustness, explainability and interpretability.

Acknowledgements

The authors would like to thank the anonymous reviewers for the constructive feedback. A special thanks to Deborah Ferreira for the helpful discussions and comments. Additionally, we would like to thank the Computational Shared Facility of the University of Manchester for providing the infrastructure to run our experiments.

References

Or Biran and Courtenay Cotton. 2017. Explanation and justification in machine learning: A survey. In *IJCAI-17 workshop on explainable AI (XAI)*, volume 8, pages 8–13.

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, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter D Turney, and Daniel Khashabi. 2016. Combining retrieval, statistics, and inference to answer elementary science questions. In *AAAI*, pages 2580–2586. Citeseer.

Peter Clark, Philip Harrison, and Niranjan Balasubramanian. 2013. A study of the knowledge base requirements for passing an elementary science test. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 37–42.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Yuwei Fang, Siqi Sun, Zhe Gan, Rohit Pillai, Shuohang Wang, and Jingjing Liu. 2020. Hierarchical graph network for multi-hop question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8823–8838.

Deborah Ferreira and André Freitas. 2020a. Natural language premise selection: Finding supporting statements for mathematical text. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 2175–2182.

Deborah Ferreira and André Freitas. 2020b. Premise selection in natural language mathematical texts. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7365–7374.

Daniel Fried, Peter Jansen, Gustave Hahn-Powell, Mihai Surdeanu, and Peter Clark. 2015. Higher-order lexical semantic models for non-factoid answer reranking. *Transactions of the Association for Computational Linguistics*, 3:197–210.

Peter Henderson, Jieru Hu, Joshua Romoff, Emma Brunskill, Dan Jurafsky, and Joelle Pineau. 2020. Towards the systematic reporting of the energy and carbon footprints of machine learning. *Journal of Machine Learning Research*, 21(248):1–43.

Peter Jansen, Niranjan Balasubramanian, Mihai Surdeanu, and Peter Clark. 2016. What's in an explanation? characterizing knowledge and inference requirements for elementary science exams. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2956–2965.

Peter Jansen, Rebecca Sharp, Mihai Surdeanu, and Peter Clark. 2017. Framing qa as building and ranking intersentence answer justifications. *Computational Linguistics*, 43(2):407–449.

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

Peter Jansen, Elizabeth Wainwright, Steven Marmorstein, and Clayton Morrison. 2018. Worldtree: A corpus of explanation graphs for elementary science questions supporting multi-hop inference. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Peter A Jansen. 2018. Multi-hop inference for sentence-level textgraphs: How challenging is meaningfully combining information for science question answering? *NAACL HLT 2018*, page 12.

Daniel Khashabi, Erfan Sadeqi Azer, Tushar Khot, Ashish Sabharwal, and Dan Roth. 2019. On the capabilities and limitations of reasoning for natural language understanding. *arXiv preprint arXiv:1901.02522*.

Daniel Khashabi, Tushar Khot, Ashish Sabharwal, Peter Clark, Oren Etzioni, and Dan Roth. 2016. Question answering via integer programming over semi-structured knowledge. In *IJCAI International Joint Conference on Artificial Intelligence*, volume 2016, pages 1145–1152.

Daniel Khashabi, Tushar Khot, Ashish Sabharwal, and Dan Roth. 2018. Question answering as global reasoning over semantic abstractions. *Conference of Association for the Advancement of Artificial Intelligence*.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. Unifiedqa: Crossing format boundaries with a single qa system. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1896–1907.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2017a. Answering complex questions using open information extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 311–316.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2017b. Answering complex questions using open information extraction. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*.

Souvik Kundu, Tushar Khot, Ashish Sabharwal, and Peter Clark. 2019. Exploiting explicit paths for multi-hop reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2737–2747.

Peter Lipton. 2017. Inference to the best explanation. *A Companion to the Philosophy of Science*, pages 184–193.

Alexander A Loboda, Maxim N Artyomov, and Alexey A Sergushichev. 2016. Solving generalized maximum-weight connected subgraph problem for network enrichment analysis. In *International Workshop on Algorithms in Bioinformatics*, pages 210–221. Springer.

Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38.

Jianmo Ni, Chenguang Zhu, Weizhu Chen, and Julian McAuley. 2019. Learning to attend on essential terms: An enhanced retriever-reader model for open-domain question answering. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 335–344.

Charles Sanders Peirce. 1960. *Collected papers of charles sanders peirce*, volume 2. Harvard University Press.

Lin Qiu, Yunxuan Xiao, Yanru Qu, Hao Zhou, Lei Li, Weinan Zhang, and Yong Yu. 2019. Dynamically fused graph network for multi-hop reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6140–6150.

Nils Reimers, Iryna Gurevych, Nils Reimers, Iryna Gurevych, Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2019. 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.

Vivian Dos Santos Silva, Siegfried Handschuh, and André Freitas. 2018. Recognizing and justifying text entailment through distributional navigation on definition graphs. In *AAAI*, pages 4913–4920.

Vivian S Silva, André Freitas, and Siegfried Handschuh. 2019. Exploring knowledge graphs in an interpretable composite approach for text entailment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7023–7030.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Kai Sun, Dian Yu, Dong Yu, and Claire Cardie. 2019. Improving machine reading comprehension with general reading strategies. *Proceedings of the 2019 Conference of the North*.

Mokanarangan Thayaparan, Marco Valentino, and André Freitas. 2020. A survey on explainability in machine reading comprehension. *arXiv preprint arXiv:2010.00389*.

Mokanarangan Thayaparan, Marco Valentino, Viktor Schlegel, and André Freitas. 2019. Identifying supporting facts for multi-hop question answering with document graph networks. In *Proceedings of the Thirteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-13)*, pages 42–51.

Marco Valentino, Mokanarangan Thayaparan, and André Freitas. 2020. Explainable natural language reasoning via conceptual unification. *arXiv preprint arXiv:2009.14539*.

Marco Valentino, Mokanarangan Thayaparan, and Andre Freitas. 2021. Unification-based reconstruction of multi-hop explanations for science questions. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, pages 200–211. 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 12th Language Resources and Evaluation Conference*, pages 5456–5473, Marseille, France. European Language Resources Association.

Vikas Yadav, Steven Bethard, and Mihai Surdeanu. 2019a. Alignment over heterogeneous embeddings for question answering. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2681–2691.

Vikas Yadav, Steven Bethard, and Mihai Surdeanu. 2019b. Quick and (not so) dirty: Unsupervised selection of justification sentences for multi-hop question answering. 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 2578–2589, Hong Kong, China. Association for Computational Linguistics.

Yuyu Zhang, Hanjun Dai, Kamil Toraman, and Le Song. 2018. Kgˆ 2: Learning to reason science exam questions with contextual knowledge graph embeddings. *arXiv preprint arXiv:1805.12393*.

A Supplementary Material

This section consists of all the hyperparameters, code and libaries used in our approach. We present this in the hope it fosters reproducibility.

A.1 Linear Programming Optimization

The components of the linear programming system is as follows:

- Solver: CPLEX optimization studio V12.9.0 https://www.ibm.com/products/ilog-cplex-optimization-studio

The hyperparatemers used in the LP constraints:

- Maximum number of abstract facts (K): 2
- Average time per epoch: 6 minutes for train-set
- Number of Epochs: 200

Infrastructures used:

- CPU Cores: 32
- CPU Model: Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz
- Memory: 128GB
- OS: Ubuntu 18.04 LTS

A.2 Parameter tuning

Our work employed Bayesian optmization with Gaussian process for hyperparamter tuning. We used the https://github.com/fmfn/BayesianOptimization: Bayesian-Optimization python library to implement the code. These parameters are as follows:

- Gaussian Kernels:
  - RationalQuadratic Kernel with default parameters
  - WhiteKernel with noise level of 1e-5, noise level bounds (1e-10, 1e1) and rest of the default parameters
- Number of iterations: 200
- alpha (α): 1e-8
- random state: 1

A.3 Sentence-BERT for Semantic Similarity Scores

We use: *roberta-large nli-stsb mean-tokens* model to calculate the semantic similarity scores.

A.4 BERT model

The BERT model was taken from the Huggingface Transformers (https://github.com/huggingface/transformers) library and fine-tuned using 4 Tesla V100 GPUs for 10 epochs in total with batch size 16 for BERT$_{Large}$ and 32 for for BERT$_{Base}$. The hyperparameters adopted for BERT are as follows:

- gradient accumulation steps: 1
- learning rate: 1e-5
- weight decay: 0.0
- adam epsilon: 1e-8
- warmup steps: 0
- max grad norm: 1.0
- seed: 42

A.5 PathNet

We use the code and dependencies provided by the PathNet github repository (https://github.com/allenai/PathNet).
We used the training config provided for OpenBookQA as a baseline: https://github.com/allenai/PathNet, file name: *blob/master/training_configs/config_obqa.json*.

A.6 Relevant facts retrieval

The code for BM25 and Unification retrieval approaches were adopted from the Unification Explanation Retrieval GitHub repository (https://github.com/ai-systems/unification_reconstruction_explanations).

A.7 Code

The code for reproducing the ExplanationLP and the experiments described in this paper are attached with the code appendix and will be available at the following GitHub repository (with a Dockerized container): https://github.com/ai-systems/explanationlp.

A.8 Data

WorldTree Dataset : The 2,290 questions in the WorldTree corpus are split into three different subsets: *train-set* (987), *dev-set* (226), and *test-set* (1,077). We only considered questions with explanations for our evaluation. The reasoning behind omitting questions without explanations was to ensure fact coverage for all questions. For AbstractKB building we excluded facts from 'KINDOF' and 'SYNONYMY' table, as these are the one primarily composed of grounding facts.

ARC-Challenge Dataset : Only used the Challenge split: https://allenai.org/data/arc.