Neural Natural Logic Inference for Interpretable Question Answering

Jihao Shi, Xiao Ding[*], Li Du, Ting Liu, and Bing Qin
Research Center for Social Computing and Information Retrieval
Harbin Institute of Technology, China
{jhshi, xding, ldu, tliu, qinb}@ir.hit.edu.cn

Abstract

Many open-domain question answering problems can be cast as a textual entailment task, where a question and candidate answers are concatenated to form hypotheses. A QA system then determines if the supporting knowledge bases, regarded as potential premises, *entail* the hypotheses. In this paper, we investigate a neural-symbolic QA approach that integrates *natural logic* reasoning within deep learning architectures, towards developing effective and yet explainable question answering models. The proposed model gradually bridges a hypothesis and candidate premises following natural logic inference steps to build proof paths. Entailment scores between the acquired intermediate hypotheses and candidate premises are measured to determine if a premise entails the hypothesis. As the natural logic reasoning process forms a tree-like, hierarchical structure, we embed hypotheses and premises in a Hyperbolic space rather than Euclidean space to acquire more precise representations. Empirically, our method outperforms prior work on answering multiple-choice science questions, achieving the best results on two publicly available datasets. The natural logic inference process inherently provides evidence to help explain the prediction process.

1 Introduction

Question answering (QA) is an important real-life NLP application but also a challenging task for assessing how well AI systems understand human language and perform reasoning to answer questions. A main challenge of QA is that the answers often do not explicitly exist in a supporting knowledge base but instead need to be *inferred* from it. Prior work (Angeli et al., 2016) has viewed QA as a textual entailment problem performed on a large premise set, where a question and candidate answers are formulated as hypotheses that need to be proved.

Neural networks have recently become the mainstream models for QA (Lukovnikov et al., 2017; Jia et al., 2018; Yang et al., 2019). Most of the models, however, are unable to give explainable inference results. Developing effective and yet explainable question answering models has attracted more attention (Abujabal et al., 2017; Yang et al., 2018; Zhou et al., 2018; Sydorova et al., 2019; Weber et al., 2019).

In this paper, we investigate a neural-symbolic QA approach that integrates *natural logic* reasoning (Lakoff, 1970; Nairn et al., 2006; MacCartney and Manning, 2009) within deep learning architectures for QA, aiming to keep the backbone of inference based on the natural logic formalism, while integrating neural networks to make the systems powerful and robust. Conventional natural logic has been designed for natural language inference and question answering (MacCartney and Manning, 2009; Angeli and Manning, 2014). As opposed to performing deduction on an abstract logical form, e.g., first-order logic (FOL) or its fragments, in which obtaining representation for abstract logic forms is known to face many thorny challenges, natural logic provides a formal proof framework based on the monotonicity calculus or projectivity.

We present the **Neu**ral **N**atural **L**ogic **I**nference (`NeuNLI`) framework for question answering. The core idea of `NeuNLI` is bridging a hypothesis and candidate premises by following natural logic inference steps and incorporating neural models to help build the proof paths. `NeuNLI` first converts a question and candidate answers to form declarative sentences, namely hypotheses. It then rewrites these original hypotheses to obtain intermediate hypotheses and repeats this process to construct a proof tree for each question-answer pair.

Since the reasoning process forms a tree-like, hierarchical structure (Angeli and Manning, 2014), it can lead to structural distortion when learning embeddings for hypotheses and premises in the

[*] Corresponding author.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 3673–3684
November 7–11, 2021. ©2021 Association for Computational Linguistics

Euclidean space (Sarkar, 2011; Sala et al., 2018). Additionally, natural language text exhibits hierarchical structure in a variety of respects (Dhingra et al., 2018). `NeuNLI` projects the question and answer embeddings to the Hyperbolic space. For a proof tree, `NeuNLI` computes an entailment score between tree nodes and candidate premises in a Hyperbolic space and use that to help select the answer. We demonstrate modelling entailment score in the Hyperbolic space improves the performance.

To train the above process in an end-to-end differentiable manner, we utilize the Gumbel-Softmax technique (Jang et al., 2017), which can effectively approximate the discrete variable, as an approximation of the non-differentiable selecting process of candidate mutations. In summary, the contributions of our work are as follows: (1) We introduce a novel framework `NeuNLI`, which combines the advantages of natural logic and deep neural networks for question answering. (2) Our proposed model provides step-by-step explanation for how the prediction was derived. (3) The proposed model achieves new state-of-the-art performance on two QA datasets. We provide detailed analyses demonstrating how the model works to achieve the improvement. The code is released at https://github.com/Shijihao/NeuNLI.

2 Background

2.1 The Problem

Consider an example from a multiple-choice science question from (Clark, 2015) and as shown in the following example.

Example–1:
Question: The main function of a fish's fins is to help the fish _____.
(A) reproduce (B) see (C) breathe (D) move

Knowledge Base: . . . A fish has a flipper or fin that helps them swim. The dorsal fin can help to keep the fish stable in the water. . . .

Given a science question, four candidate answers, and relevant knowledge, a model needs to choose the correct answer supported by the knowledge base. Following Clark et al. (2018), we explore to solve the multiple-choice question answering as a textual entailment problem. Specifically, a question and four candidate answers can be converted to four declarative sentences, namely target hypothesis h_i where $i \in \{1, 2, 3, 4\}$. We will retrieve relevant knowledge, a premise set $P = \{p_1, \ldots, p_j, \ldots, p_k\}$, from the knowledge base and determine one that entails one of the four hypotheses, where k represents the number of supporting premises. Central to our approach is the development of neural-symbolic model that uses natural logic as the backbone prover and leverages the expressiveness of neural models to help construct this proving process.

Relation	**Name**	**Example**
$x \equiv y$	equivalence	garbage $\equiv$ rubbish
$x \sqsubseteq y$	forward entailment	dog $\sqsubseteq$ animal
$x \sqsupseteq y$	reverse entailment	animal $\sqsupseteq$ dog
$x \curlywedge y$	negation	usual $\curlywedge$ unusual
$x \downharpoonleft\upharpoonright y$	alternation	monkey $\downharpoonleft\upharpoonright$ elephant
$x \smile y$	cover	mammal $\smile$ nonhuman
$x \# y$	independence	angry $\#$ fridge

Table 1: Seven logic relations.

2.2 Natural Logic

Natural Logic (Lakoff, 1970) is a formal proof theory built on the syntax of human language, which can be traced to the syllogisms of Aristotle. It aims to capture logical inferences by appealing directly to the structure of language. Specifically, the logical inferences are directly operated on the surface form of language based on the monotonicity calculus or projectivity (MacCartney and Manning, 2009; Valencia, 1991), as opposed to running deduction on an abstract logical form, first-order logic (FOL), or its *fragments*. For natural language, obtaining a representation of abstract logic forms is known to face many thorny challenges. In this research, we investigate developing neural natural logic models for QA, which provide insight into the derivation process but also sidestep the difficulties of translating sentences into FOL.

Natural logic proving is operated by inserting, deleting, or mutating words following monotonicity calculus or projectivity (MacCartney and Manning, 2009; Valencia, 1991). In their recent work MacCartney and Manning (2009) utilize seven logical relations as shown in Table 1. For example, mutating *animals* to *dogs* corresponds to a reverse entailment relation, i.e., *animals* $\sqsupseteq$ *dogs*. Natural logic then projects the lexical relation based on the monotonicity or projectivity determined by the context. According to the monotonicity calculus, upward monotone preserves the logical relation, while downward monotone can change the logical

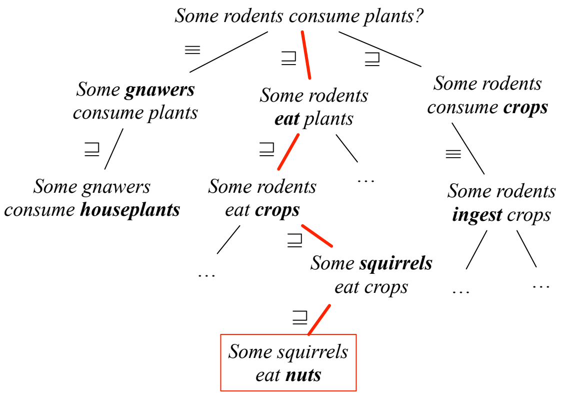
Some rodents consume plants?
≡
⊒
⊒
Some **gnawers** consume plants
Some rodents **eat** plants
Some rodents consume **crops**
⊒
⊒
≡
Some gnawers consume **houseplants**
Some rodents eat **crops**
…
Some rodents **ingest** crops
⊒
Some **squirrels** eat crops
…
…
…
⊒
Some squirrels eat **nuts**

Figure 1: Natural logic proof process. It starts from a hypothesis *rodents consume plants* and finds out a premise *squirrels eat nuts*. Labels on the edges show the logical relations between associated sentences.

relation. For example, the quantifier *all* has a downward monotone in its first argument. Accordingly, given *animals* ⊒ *dogs*, we know *all animals* ⊑ *all dogs* (e.g., as in *all animals need water* ⊑ *all dogs need water*).

2.3 Natural Logic Inference

Natural logic inference casts inference as a search problem: given a hypothesis and an arbitrarily large corpus of text, it searches through the space of lexical mutations (e.g., *eat* → *consume*), with associated costs, until a premise is found (Angeli and Manning, 2014). The entire inference process, constructed in reverse, starts from the hypothesis. An example search using natural logic inference is given in Figure 1. The root denotes one of the hypotheses in our task, and the relations along the edges denote relations between the associated sentences.

3 Method

In this paper we propose the **Neu**ral **N**atural **L**ogic **I**nference (`NeuNLI`) framework, aiming to combine the advantages of natural logic and deep neural networks for question answering, which builds explainability in the model and leverages the powerful capacity and robustness of neural models. Figure 2 depicts the overall architecture of `NeuNLI`; the pseudocode of `NeuNLI` is listed in Algorithm 1. In the following subsections, we discuss `NeuNLI` in detail.

As the starting point, given a question sentence, "*In New York State, the longest period of daylight occurs during which month*" and candidate answers, `NeuNLI` converts the question and each answer (say, "*June*") to a declarative hypothesis sentence h_i, i.e., "*In New York state, the longest period of daylight occurs during June*".

Algorithm 1: `NeuNLI` Pseudocode

```
Input: Hypothesis $h_i$, premises
    $P = \{p_1, \ldots, p_j, \ldots, p_k\}$, maximum
    iteration $i_{max}$
Output: Entailment score $s_i$
Initialization: $s_i \leftarrow 0$
$H_{cand} \leftarrow Insertion\_Deletion\_Mutation(h_i)$ ;
$S_{cj} \leftarrow Score(h_c, p_j) \forall h_c \in H_{cand}, p_j \in P$;
$s_i \leftarrow max(S_{cj})$;
while $iteration < i_{max}$ do
    rank $H_{cand}$ according to $S_{cj}$ in descending
      order;
    $H_{cand} \leftarrow H_{cand}[: top_k]$ ;
    $H'_{cand} \leftarrow []$ ;
    for $h_c$ in $H_{cand}$ do
        add $Insertion\_Deletion\_Mutation(h_c)$
          to $H'_{cand}$ ;
    end
    $S_{ij} \leftarrow Score(h'_i, p_j) \forall h'_i \in H'_{cand}, p_j \in P$;
    $s^*_i \leftarrow max(S_{ij})$;
    if $s^*_i > s_i$ then
        $s_i \leftarrow s^*_i$;
    end
    $H_{cand} \leftarrow H'_{cand}$
end
Return: $s_i$
```

3.1 Candidate Premises Retrieval

The knowledge base $\mathcal{K}$ consists of unstructured text. This makes available the great amount of text as knowledge source to help perform question answering. Given a hypothesis, as shown in the right part of Figure 2, `NeuNLI` first retrieves candidate premises. Specifically, a premise is one of the sentences in the knowledge base $\mathcal{K} = \{p_1, \ldots, p_n\}$. Given a hypothesis h_i, we obtain the representation of h_i and each p_j in $\mathcal{K}$ by computing the average Glove word embeddings (Pennington et al., 2014) of it, respectively. Then we calculate the cosine similarity between h_i and each p_j in $\mathcal{K}$, respectively, to find the top k relevant candidate premises (k is tuned on the development set).

3.2 Contextualized Neural Natural Logic Prover

Candidate Proof Path Generation. As shown in Figure 2, starting from the hypothesis at the root, the backward proof process needs to generate proof paths to help find supporting premises from the candidate premise pool retrieved above. The paths are built by adding intermediate hypotheses, following natural logic inference steps and utilizing neural

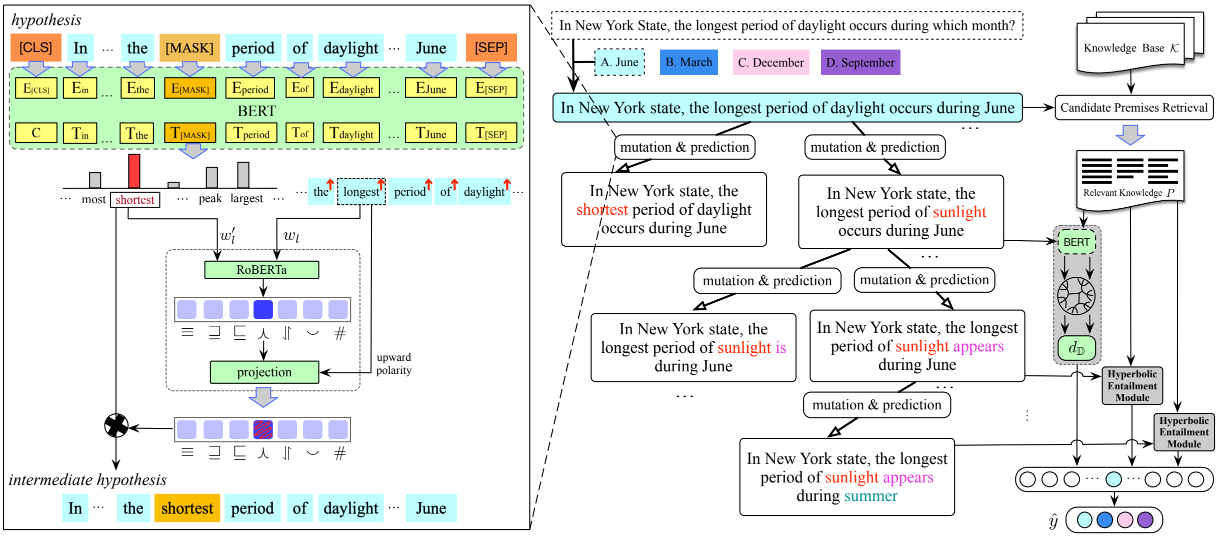
hypothesis
[CLS]
In
the
[MASK]
period
of
daylight
June
[SEP]
E[CLS]
Ein
Ethe
E[MASK]
Eperiod
Eof
Edaylight
EJune
E[SEP]
BERT
C
Tin
Tthe
T[MASK]
Tperiod
Tof
Tdaylight
TJune
T[SEP]
most
shortest
peak
largest
the
longest
period
of
daylight
w'_l
w_l
RoBERTa
≡ ⊐ ⊏ ⅄ ⥯ ⌣ #
projection
upward polarity
intermediate hypothesis
In
the
shortest
period
of
daylight
June
In New York State, the longest period of daylight occurs during which month?
A. June
B. March
C. December
D. September
In New York state, the longest period of daylight occurs during June
mutation & prediction
In New York state, the shortest period of daylight occurs during June
In New York state, the longest period of sunlight occurs during June
In New York state, the longest period of sunlight is during June
In New York state, the longest period of sunlight appears during June
In New York state, the longest period of sunlight appears during summer
Knowledge Base $\mathcal{K}$
Candidate Premises Retrieval
Relevant Knowledge P
BERT
$d_{\mathbb{D}}$
Hyperbolic Entailment Module
Hyperbolic Entailment Module
$\hat{y}$

Figure 2: Overview of the proposed NeuNLI framework for Question Answering.

networks to suggest candidates. An intermediate hypothesis always entails the original hypothesis—if we can find a premise entailing an intermediate hypothesis, we also prove the original hypothesis.

Intuitively, there is no need to mutate each word in a hypothesis. Thus, we first find out function words in advance. For each word w_i in a hypothesis, we use the NLTK (Bird et al., 2009) toolkit to obtain its *part of speech* tag $w_{i_{pos}}$ and apply rules to filter out words that have little influence on the semantics of the hypothesis. Words with the following part of speech will be neglected: preposition, determiner, coordinating conjunction, cardinal numbers, personal pronoun, and modal verb. Also, punctuation words and stop words will be excluded. Subsequently, we conduct inference starting from the original hypothesis h_i that consists of L words, $h_i = (w_1^i, \dots, w_l^i, \dots, w_L^i)$. We first mask a word in the hypothesis and then feed it into BERT to predict the masked token as shown in the upper-left subfigure of Figure 2.

The probability of the word w on the l'th position of h_i with parameters θ is defined by:

$$p_l(w|\theta) = w_o^T f_\theta(S_{\backslash l}), \qquad (1)$$

where w_o is the one-hot vector for the word w on the l'th position; $f_\theta(\cdot)$ is a multi-layer bidirectional transformer model (Vaswani et al., 2017) and $S_{\backslash l} = (w_1^i, ..., w_{l-1}^i, \texttt{[MASK]}, w_{l+1}^i, ...w_L^i)$. In order to narrow the semantic distance, the reversed search also deal with lexical insertion and deletion. For example, the sentence *some grey squirrels eat nuts* would entail *some squirrels eat nuts* by lexical insertion. As we know, deleting noun (or verb) is very likely to result in incomplete sentences, whereas inserting noun (or verb) does not guarantee the resulting sentences conform to the grammar. Thus, we choose to only insert (or delete) adjective, to conduct inference. Generating candidate words for insertion also utilizes the mask mechanism: we insert a mask in front of the corresponding noun. The position of insertion and deletion will be tagged to avoid repetitive insertion/deletion operations at the same location.

Due to the nature of masked language modeling, we take advantage of the mask mechanism for lexical mutation. In this way, the context of the mutated word w_l can be considered. According to the probability $p_l(w|\theta)$, the candidate words can be ranked in descending order. The higher the probability, the more relevant the candidate word w'_l to the original word w_l. So far, we have obtained a list of candidate words.

Proof Path Filtering. The mask mechanism does not guarantee semantic coherence to the original hypothesis as shown in the left part of Figure 2. The original hypothesis is "*. . . the longest period of daylight . . .* ". Through the mutation of the word "*longest*", the candidate words may contain the word "*shortest*", which fits well into the grammar and context of the sentence but changes the semantics of the original hypothesis. To keep a high semantic similarity, we need to judge whether the mutation operation would change the semantics of the original hypothesis using a logical relation prediction module, and filter out the incorrect mutations. Here, the candidate word w'_l is "*short-*

r	$\equiv$	$\sqsubseteq$	$\sqsupseteq$	$\curlywedge$	$\downharpoonleft\upharpoonright$	$\smile$	$\#$
$\phi(r)$	$\equiv$	$\sqsupseteq$	$\sqsubseteq$	$\downharpoonleft\upharpoonright$	$\curlywedge$	$\smile$	$\#$

Table 2: The projection function ϕ when the lexical polarity of the mutated word is downward. The input r is the predicted lexical relation between the mutated word and the mutating word. Note that the projection function ϕ is the identity function when the lexical polarity of the mutated word is upward.

est", while the mutated word w_l is "*longest*". First, we use the fine-tuned RoBERTa (Liu et al., 2019) to predict the logical relation between "*shortest*" and "*longest*". The input form of the RoBERTa is `[CLS]` w_l `[SEP]` w'_l `[SEP]`, where w_l is assigned to segment 0 and w'_l is assigned to segment 1. The predicted result is negation relation ($\curlywedge$), calculated by the representation of the `[CLS]` token.

Then, we use the projection function ϕ to obtain the sentence-level semantic relation according to the predicted lexical relation and the lexical polarity of the word w_l. If the lexical polarity is upward, the sentence-level relation will be identical to the lexical relation. Otherwise, the projection from the word-level relation to the sentence-level relation is performed as shown in Table 2. We employ Stanford *natlog* parser (Manning et al., 2014) to acquire the lexical polarity of words. For example, as the polarity of the mutated word "*longest*" is upward, and the logical relation between "*longest*" and "*shortest*" is $\curlywedge$, the semantic relation of the hypothesis h_i and the intermediate hypothesis h'_i still maintains $\curlywedge$. If the predicted polarity of "*longest*" is downward, the sentence-level relation will be $\downharpoonleft\upharpoonright$. As we only conduct inference on the sentence-level relation of $\equiv$ or $\sqsupseteq$, this mutation would be filtered out.

Entailment Score Estimation in Hyperbolic Space. Given the intermediate hypothesis h'_i, we need to calculate the entailment score s_{ij} between the intermediate hypothesis h'_i and each candidate premise p_j. The representation of intermediate hypothesis and candidate premise in Euclidean space is calculated by BERT model with the input `[CLS]` p_j `[SEP]` h'_i `[SEP]`. We take the embedding of token `[CLS]` and middle token `[SEP]` as the representation of the candidate premise and the intermediate hypothesis, denoted as $\boldsymbol{v}^E_{p_j}$ and $\boldsymbol{v}^E_{h'_i}$.

For the tree-like, hierarchical structure constructed by the reasoning process, the number of intermediate hypotheses grows exponentially. However, the Euclidean space grows polynomially, which would lead to structural distortion in the Euclidean space (Sarkar, 2011; Sala et al., 2018). Additionally, natural language text itself exhibits hierarchical structure. Thus, we calculate the entailment scores between them in Hyperbolic space as shown in the right part of Figure 2. Here, we choose the Poincaré ball model (Cannon et al., 1997) to project the candidate premise and intermediate hypothesis into the Hyperbolic space to acquire more precise representations. We exploit the re-parameterization technique (Dhingra et al., 2018; López et al., 2019; Cao et al., 2020) to implement it, which involves calculating a direction vector $\mathbf{m}$ and a norm magnitude μ. Take $v^E_{p_j}$ as an example to illustrate the procedure:

$$\begin{aligned} \overline{\mathbf{m}}_{p_j} &= \psi_{\text{dir}}\left(\boldsymbol{v}^E_{p_j}\right), & \mathbf{m}_{p_j} &= \frac{\overline{\mathbf{m}}_{p_j}}{\|\overline{\mathbf{m}}_{p_j}\|} \\ \bar{\mu}_{p_j} &= \psi_{\text{norm}}\left(\boldsymbol{v}^E_{p_j}\right), & \mu_{p_j} &= \sigma\left(\bar{\mu}_{p_j}\right) \end{aligned} \tag{2}$$

where $\psi_{dir} : \mathbb{R}^d \rightarrow \mathbb{R}^{d_H}$ is a multi-layer perceptron. $\psi_{norm} : \mathbb{R}^d \rightarrow \mathbb{R}$ is a linear function. σ is the sigmoid function to ensure the resulting norm $\mu_{p_j} \in (0, 1)$. The re-parameterized premise representation is defined as $\boldsymbol{v}^H_{p_j} = \mu_{p_j}\mathbf{m}_{p_j}$, which lies in Hyperbolic space $\mathcal{B}^{d_H}$. The re-parameterization technique has the ability to avoid the need to adopt the stochastic Riemannian optimization method (Bonnabel, 2013). Instead, we can exploit AdamW (Loshchilov and Hutter, 2019) to update the parameters in the entire model.

The entailment score in Hyperbolic space is calculated by the hyperbolic distance:

$$\begin{aligned} & d_{\mathbb{D}}(\boldsymbol{v}^H_{p_j}, \boldsymbol{v}^H_{h'_i}) \\ & = \cosh^{-1}(1 + 2\frac{\|\boldsymbol{v}^H_{p_j} - \boldsymbol{v}^H_{h'_i}\|^2}{(1 - \|\boldsymbol{v}^H_{p_j}\|^2)(1 - \|\boldsymbol{v}^H_{h'_i}\|^2)}), \end{aligned} \tag{3}$$

where $\boldsymbol{v}^H_{p_j}$ and $\boldsymbol{v}^H_{h'_i}$ are representations of the candidate premise and intermediate hypothesis in Hyperbolic space. We then utilize a learnable classifier to project $d_{\mathbb{D}}(\boldsymbol{v}^H_{p_j}, \boldsymbol{v}^H_{h'_i})$ to a scalar entailment score s_{ij}. The maximum entailment score $s_i = \max_j(s_{ij})$ is used as the supporting probability to the hypothesis h_i, i.e. the probability of the corresponding answer. This is repeated for all answers, and the answer with the highest entailment score $\max_i(s_i)$ is selected as the correct answer.

3.3 Gumbel-Softmax Training

Note that the above learning process is not differentiable and the training signal cannot be passed to the parameters of pre-trained language model. To address this, we adopt the Gumbel-Softmax technique (Jang et al., 2017) to train the whole process in an end-to-end manner. Gumbel-Softmax technique has been shown an effective approximation to the discrete variable. Therefore, we use

$$\mathbf{w}_j = \frac{\exp((\log(p_l(w_j|\theta)) + g_j)/\tau)}{\sum_i \exp((\log(p_l(w_i|\theta)) + g_i)/\tau)} \quad (4)$$

as the approximation of the one-hot vector of a selected mutating word on the l'th position, where w_i is the i'th token that appears in the vocab of BERT model. g_j are i.i.d samples drawn from Gumbel(0,1) [1] and τ is a constant that controls the smoothness of the distribution.

3.4 Objective Function

We normalize prediction scores across all candidate answers using the softmax function and train the model using the cross-entropy loss:

$$\hat{y_i} = \frac{\exp(s_i)}{\sum_{n=1}^{C} \exp(s_n)}, \quad (5)$$

$$\mathcal{L} = -\sum_{i=1}^{C} t_i \log(\hat{y_i}), \quad (6)$$

where C is the number of candidate answers. s_i is the entailment score corresponding to the answer i and t_i is 1 when the i'th candidate answer is correct, otherwise t_i is 0. We minimize the cross-entropy loss between the prediction result and the ground truth.

4 Experiment Set-Up

Datasets, Baselines, and Implementation Details. We evaluate the performance of our model on two publicly available datasets (Angeli et al., 2016). Both datasets are made up of non-diagram multiple-choice science questions from the New York Regents 4^{th} Grade Science Exams (NYSED, 2014). We use the same datasets (QA-S and QA-L) and knowledge bases (Barron's and SCITEXT) as the baseline (Angeli et al., 2016). The details of datasets and knowledge bases can be found in Appendix A. We compare NeuNLI with **Solr**, **Classifier** (Angeli et al., 2016), **Evaluation Function**, **NaturalLI** (Angeli et al., 2016), **HyperQA** (Tay et al., 2018), **SemBERT** (Zhang et al., 2020) and **NeuNLI-E**. Descriptions of the baseline methods are detailed in Appendix B. Additionally, experiment settings are further discussed in Appendix C.

$\equiv$	$\sqsubseteq$	$\sqsupseteq$	$\curlywedge$	$\downharpoonleft\upharpoonright$	$\smile$	$\#$
10,000	10,000	10,000	1,320	10,000	1,650	10,000

Table 3: Statistics of lexical relation prediction corpus.

Construction of Lexical Relation Prediction Corpus. To better predict lexical relations between the original word and the candidate mutating word, we build a set of lexical pairs to train the prediction model. These lexical pairs are built upon the lexical knowledge base WordNet (Miller, 1992). We regard words in the same synsets of the WordNet as having the equivalence relation $\equiv$. Words with hypernymy and hyponymy relations in the WordNet are cast as having the forward $\sqsubseteq$ and reverse $\sqsupseteq$ entailment relation, respectively. The antonymy relation in the WordNet can be naturally projected as the negation relation $\curlywedge$ of the natural logic. For a synset in WordNet, the relation between its hypernyms (or between its hyponyms) is regarded as the alternation relation $\downharpoonleft\upharpoonright$ in natural logic. Besides, for a synset in WordNet, its hyponymy and its antonym have the cover relation $\smile$ in natural logic. As for the independence relation $\#$ in natural logic, we randomly extract lexical pairs from the WordNet and then filter out pairs that have the other six lexical relations and the rest can be regarded as the independence relation.

The number of seven lexical relations in natural logic is shown in Table 3. We split the number of each relation with the ratio of 8:1:1 to fine-tune a pre-trained language model.

5 Experiment Results

We list the test accuracy of baseline methods and NeuNLI on two test sets in Table 4 (QA-S) and Table 5 (QA-L), respectively. In Table 4, we also present results utilizing two different knowledge bases. We find that:

(1) Compared with NaturalLI (Angeli et al., 2016), our method performs better because we consider the contextual information during the natural logic-based reasoning process. This helps to reduce the unnecessary expansion of irrelevant lexical mutation and make NeuNLI focusing on the right

[1] We sample g by drawing $u \sim$ Uniform(0, 1) then computing $g = -log(-log(u))$

Model	**Barron's**	**SCITEXT**
Solr Only	42	58
Classifier	52	60
+ Solr	48	64
Evaluation Function	54	63
+ Solr	45	58
NaturalLI (Angeli et al., 2016)	51	61
+ Solr	49	61
+ Solr + Classifier	49	67
HyperQA (Tay et al., 2018)	54	62
SemBERT (Zhang et al., 2020)	53	59
NeuNLI-E (Ours)	57	67
NeuNLI (Ours)	**64***	**72***

Table 4: Accuracy (%) on the QA-S test set with 68 examples. The results are shown in integer form as (Angeli et al., 2016). Bold denotes best results.* denotes a significance test at the level of 0.05.

Model	**Accuracy**
Solr Only	46.8
Classifier	43.6
NaturalLI (Angeli et al., 2016)	46.4
+ Solr	48.0
HyperQA (Tay et al., 2018)	47.6
SemBERT (Zhang et al., 2020)	47.2
NeuNLI-E (Ours)	48.8
NeuNLI (Ours)	**50.8***

Table 5: Accuracy (%) on the QA-L test set with 250 examples. Bold denotes the best result. * denotes a significance test at the level of 0.05.

Relation	**BERT**			**RoBERTa**		
	P	R	F1	P	R	F1
equivalence	0.79	0.83	0.81	0.81	0.85	0.83
forward entailment	0.75	0.69	0.72	0.75	0.74	0.75
reverse entailment	0.69	0.69	0.69	0.70	0.72	0.71
negation	0.74	0.58	0.65	0.85	0.63	0.72
alternation	0.54	0.64	0.58	0.57	0.61	0.59
cover	0.42	0.32	0.36	0.48	0.32	0.39
independence	0.63	0.58	0.60	0.66	0.62	0.64

Table 6: Performance of lexical relation prediction.

reasoning path.

(2) Comparison between HyperQA (Tay et al., 2018) and `NeuNLI` shows that natural logic-powered neural networks can achieve better performance on the QA datasets. Moreover, the process of natural logic reasoning can serve as the explanation of the results, while HyperQA can hardly give a reasonable explanation for its results.

(3) Our method also performs better than SemBERT (Zhang et al., 2020). Both approaches incorporate contextual semantic information with BERT for QA. In comparison, we involve natural logic for achieving this goal, which is the main reason for the improvements.

(4) `NeuNLI` outperforms NeuNLI-E mainly because we learn embeddings of the candidate premise and hypothesis in Hyperbolic space, which can acquire more precise representations.

(5) `NeuNLI` achieves the best results on the test set with two different knowledge bases: Barron's and SCITEXT. We also notice that the model with a larger knowledge base SCITEXT can achieve a better performance, which coincides with human intuition that with more knowledge, we can choose more correct answers.

(6) The experimental results on the QA-L test set in Table 5 are consistent with those on the QA-S test set in Table 4, which shows the generalization of our approach.

Precision of Lexical Relation Prediction. As the lexical relation prediction is an important module in `NeuNLI` and can affect the performance of `NeuNLI`, we evaluate the performance of this module and show the results in Table 6. We compare two pre-trained language models (BERT and RoBERTa) for lexical relation prediction. The performance of RoBERTa is better than BERT because RoBERTa utilizes a dynamic mask mechanism, which can learn more knowledge.

Human Evaluation for Explainability. We quantitatively evaluate the explainability of our model through human evaluations. Specifically, we evaluate `NeuNLI` on the QA-S dataset with the Barron's knowledge base. We employ three graduate students that majored in natural language processing to give a score belonging to {0, 1, 2} to evaluate whether the inference path derived by our model is reasonable. The semantic between the final intermediate hypothesis and the premise is irrelevant, then the score is tagged 0. The semantic between the two is very close, then tagged 2. If the gap between the two needs evaluators to imagine a context, then tagged 1. For comparison, we set NaturalLI (Angeli et al., 2016) as the baseline and the significance test is conducted using paired t-test at a significance level of 0.05. The average scores are shown in Table 7 and the significance difference is less than 0.05.

We can observe that the score of `NeuNLI` is significantly higher than that of NaturalLI. This is mainly because `NeuNLI` can generate more reasonable words by incorporating contextual semantic information into the natural logic inference process.

	NaturalLI	NeuNLI
Avg. Explainability Score	1.09	1.31*

Table 7: Average explainability score of NaturalLI and NeuNLI. * denotes a significance test at the level of 0.05.

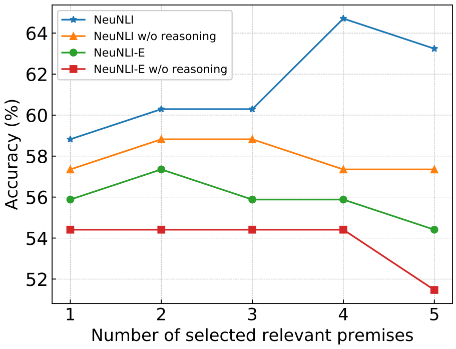

Figure 3: Ablation study by removing the main components, where "w/o" indicates without. Accuracy with different numbers of relevant premises on 68 test examples with the Barron's knowledge base. Our model NeuNLI performs the state-of-the-art.

For example, the hypothesis is "*in order to **survive**, all animals need food, water and air*". By lexical mutation in NeuNLI, we get the sentence "*in order to **live**, all animals need food, water and air*", which is closer to the premise "*animals need air, water, and food in order to live and thrive*".

Ablation Study. We conduct the ablation study on the QA-S test set with the Barron's knowledge base. The experimental results are shown in Figure 3. From the figure, we can observe that:

(1) **Effect of Number of Relevant Premises.** The accuracy continues to increase as the number of relevant premises increases from 1 to 4 in Figure 3. This is mainly because the more knowledge is involved in the model, the better performance can be achieved. While when the number of relevant premises exceeds 4, the accuracy starts to decrease, as there may be noise information included in the model by the retrieval method.

(2) **Effectiveness of Natural Logic-based Reasoning.** Comparing NeuNLI with NeuNLI w/o reasoning, we can find the performance improves significantly. The accuracy score improves from 57.35% to 64.71% on the QA-S test set with the Barron's knowledge base (setting the number of relevant premises is 4). The same conclusion can be drawn from the comparison between NeuNLI-E and NeuNLI-E w/o reasoning. It indicates that exploiting natural logic-based reasoning is very effective for QA.

6 Related Work

Question answering systems that integrate deep learning methods have made great progress in recent years (Lukovnikov et al., 2017; Bhandwaldar and Zadrozny, 2018; Jia et al., 2018; Yang et al., 2019). Many works first adopt learnable encoders for sentence representation like convolutional encoders (Zhang et al., 2017), recurrent encoders (Tay et al., 2017) and transformers (Yang et al., 2019). Then an interaction layer is devised to calculate the semantic similarity, which is the main difference in many models. Severyn and Moschitti (2015) utilize a multi-layered perceptron to combine the CNN encoded representations. Yang et al. (2016) perform a soft-attention alignment to measure word similarity between the question and the answer.

Though neural networks-based models make great advances in QA, they are short of illustrating the step-by-step prediction derivation process, where the logic-based method is adept (Rocktäschel and Riedel, 2017; Weber et al., 2019; Minervini et al., 2020), which differs from the widely used attention mechanism (Doshi-Velez and Kim, 2017; Jain and Wallace, 2019). Angeli et al. (2016) proposed a Natural Logic Inference framework to utilize natural logic to conduct interpretable question answering and viewed the open-domain question answering as a textual entailment problem. Our NeuNLI is inspired by natural logic inference but can achieve better performance by modeling the contextual information during natural logic proving using two pre-trained language models and training the whole process in an end-to-end fashion.

7 Conclusion

In this work, we explore the feasibility of combining natural logic with neural networks for interpretable question answering. We present an end-to-end differentiable method for learning the parameters as well as the structure of natural logical rules, which is capable of considering the contextual information while conducting natural logic-based reasoning. Experimental results on the Regents Science Exam of the Aristo dataset show that our proposed model could bring improvements over baseline methods.

Acknowledgements

We thank the anonymous reviewers for their constructive comments, and gratefully acknowledge the support of the National Key Research and Development Program of China (2020AAA0106501), and the National Natural Science Foundation of China (61976073).

References

Abdalghani Abujabal, Rishiraj Saha Roy, Mohamed Yahya, and Gerhard Weikum. 2017. Quint: Interpretable question answering over knowledge bases. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 61–66.

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, Doha, Qatar. Association for Computational Linguistics.

Gabor Angeli, Neha Nayak, and Christopher D. Manning. 2016. Combining natural logic and shallow reasoning for question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 442–452, Berlin, Germany. Association for Computational Linguistics.

Abhishek Bhandwaldar and Wlodek Zadrozny. 2018. UNCC QA: Biomedical question answering system. In *Proceedings of the 6th BioASQ Workshop A challenge on large-scale biomedical semantic indexing and question answering*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Silvere Bonnabel. 2013. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229.

James W Cannon, William J Floyd, Richard Kenyon, Walter R Parry, et al. 1997. Hyperbolic geometry. *Flavors of geometry*, 31:59–115.

Pengfei Cao, Yubo Chen, Kang Liu, Jun Zhao, Shengping Liu, and Weifeng Chong. 2020. HyperCore: Hyperbolic and co-graph representation for automatic ICD coding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3105–3114, Online. Association for Computational Linguistics.

Peter Clark. 2015. Elementary school science and math tests as a driver for AI: take the aristo challenge! In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 4019–4021. AAAI Press.

Peter Clark, Niranjan Balasubramanian, Sumithra Bhakthavatsalam, Kevin Humphreys, Jesse Kinkead, Ashish Sabharwal, and Oyvind Tafjord. 2014. Automatic construction of inference-supporting knowledge bases. In *4th Workshop on Automated Knowledge Base Construction (AKBC)*.

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*.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Bhuwan Dhingra, Christopher Shallue, Mohammad Norouzi, Andrew Dai, and George Dahl. 2018. Embedding text in hyperbolic spaces. In *Proceedings of the Twelfth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-12)*, pages 59–69, New Orleans, Louisiana, USA. Association for Computational Linguistics.

Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

K. Cahill J. Barry. 2007. *Barrons Fourth Grade Science Study guide*.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not Explanation. 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 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Zhen Jia, Abdalghani Abujabal, Rishiraj Saha Roy, Jannik Strötgen, and Gerhard Weikum. 2018. TEQUILA: temporal question answering over knowledge bases. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, pages 1807–1810. ACM.

George Lakoff. 1970. Linguistics and natural logic. *Synthese*, 22(1-2):151–271.

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

Federico López, Benjamin Heinzerling, and Michael Strube. 2019. Fine-grained entity typing in hyperbolic space. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 169–180, Florence, Italy. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Denis Lukovnikov, Asja Fischer, Jens Lehmann, and Sören Auer. 2017. Neural network-based question answering over knowledge graphs on word and character level. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 1211–1220. ACM.

Bill MacCartney and Christopher D. Manning. 2009. An extended model of natural logic. In *Proceedings of the Eight International Conference on Computational Semantics*, pages 140–156, Tilburg, The Netherlands. Association for Computational Linguistics.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

George A. Miller. 1992. WordNet: A lexical database for English. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*.

Pasquale Minervini, Matko Bosnjak, Tim Rocktäschel, Sebastian Riedel, and Edward Grefenstette. 2020. Differentiable reasoning on large knowledge bases and natural language. In *AAAI*.

Rowan Nairn, Cleo Condoravdi, and Lauri Karttunen. 2006. Computing relative polarity for textual inference. In *Proceedings of the Fifth International Workshop on Inference in Computational Semantics (ICoS-5)*.

NYSED. 2014. Grade 4 elementary-level science test. `http://www.nysedregents.org/Grade4/Science/home.html`.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Tim Rocktäschel and Sebastian Riedel. 2017. End-to-end differentiable proving. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 3788–3800.

Frederic Sala, Christopher De Sa, Albert Gu, and Christopher Ré. 2018. Representation tradeoffs for hyperbolic embeddings. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 4457–4466. PMLR.

Rik Sarkar. 2011. Low distortion delaunay embedding of trees in hyperbolic plane. In *International Symposium on Graph Drawing*, pages 355–366. Springer.

Aliaksei Severyn and Alessandro Moschitti. 2015. Learning to rank short text pairs with convolutional deep neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015*, pages 373–382. ACM.

Alona Sydorova, Nina Poerner, and Benjamin Roth. 2019. Interpretable question answering on knowledge bases and text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4943–4951.

Yi Tay, Minh C. Phan, Anh Tuan Luu, and Siu Cheung Hui. 2017. Learning to rank question answer pairs with holographic dual LSTM architecture. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pages 695–704. ACM.

Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Hyperbolic representation learning for fast and efficient neural question answering. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018*, pages 583–591. ACM.

Víctor Manuel Sánchez Valencia. 1991. *Studies on natural logic and categorial grammar*. Universiteit van Amsterdam.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Leon Weber, Pasquale Minervini, Jannes Münchmeyer, Ulf Leser, and Tim Rocktäschel. 2019. NLProlog:

Reasoning with weak unification for question answering in natural language. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6151–6161, Florence, Italy. Association for Computational Linguistics.

Liu Yang, Qingyao Ai, Jiafeng Guo, and W. Bruce Croft. 2016. anmm: Ranking short answer texts with attention-based neural matching model. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016*, pages 287–296. ACM.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-end open-domain question answering with BERTserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 72–77, Minneapolis, Minnesota. 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.

Xiaodong Zhang, Sujian Li, Lei Sha, and Houfeng Wang. 2017. Attentive interactive neural networks for answer selection in community question answering. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3525–3531. AAAI Press.

Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2020. Semantics-aware BERT for language understanding. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9628–9635. AAAI Press.

Mantong Zhou, Minlie Huang, and Xiaoyan Zhu. 2018. An interpretable reasoning network for multi-relation question answering. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2010–2022.

A Appendix: Datasets

We evaluate the performance of our model on two publicly available datasets (Angeli et al., 2016). One (denoted as QA-S) consists of 108 examples in the training set, 61 examples in the validation set, and 68 examples in the test set. The other (denoted as QA-L) is larger with 500 examples, 249 examples, and 250 examples in the training, validation, and test set, respectively. Two knowledge bases are available for supporting the question answering. One is Barron's study guide (J. Barry, 2007), consisting of 1,200 sentences. The other is the SCITEXT corpus (Clark et al., 2014), which extends Barron's study guide with simple Wikipedia, dictionaries, and a science textbook, including 1,316,278 sentences.

B Appendix: Baselines

We compare NeuNLI with:

- **Solr** is an information retrieval system, which can return a confidence score according to the query. Given a hypothesis, The maximum confidence score of search results is regarded as the score for that hypothesis.
- **Classifier** (Angeli et al., 2016) is a feature-based entailment classifier, which utilizes 5 unlexicalized real-valued features. Also, the confidence score calculated by the Solr information system can be seen as an optional feature.
- **Evaluation Function** is a variation of the Classifier method. Evaluation Function uses keywords as one of the features while Classifier uses key phrases as the features.
- **NaturalLI** (Angeli et al., 2016) utilizes natural logic for question answering. They use WordNet to guide the lexical mutation process, while in our work, we adopt neural networks to conduct the lexical mutation process.
- **HyperQA** (Tay et al., 2018) learns the question and answer embeddings in Hyperbolic space. we train the model in our datasets using the same settings with NeuNLI.
- **SemBERT** (Zhang et al., 2020) incorporates explicit contextual semantic information with BERT for question answering. SemBERT is a top performer on SNLI[2], and we train SemBERT in our datasets using the same settings with NeuNLI.
- **NeuNLI-E** learns the distributed embedding representations of the candidate premise and hypothesis in Euclidean space.

C Appendix: Implementation Details

We use the base size of pre-trained language models (i.e. BERT-base and RoBERTa-base) in this paper. The dimension of vector d in Euclidean space is 768. The dimension of vector d_H in Hyperbolic space is 64. When searching for the relevant premises, we use pre-trained 300-dimensional 840B GloVe vectors (Pennington et al., 2014). During the natural logic-based reasoning, we limit the maximum searching depth is 10, and restrict the number of relevant premises to be no more than 5. In the re-parameterization technique, The number of hidden layer of the multi-layer perceptron is 1 and the dimension of the hidden layer is 384. The initial learning rate is selected from $\{$1e-5, 5e-5, 3e-6$\}$. The dropout rate is 0.3. Our model is trained on one Tesla V100 GPU. For all experiments, we pick the model which works best on the validation set and then evaluate it on the test set. We use the default hyper-parameters as initial and fine-tune the pre-trained model (Devlin et al., 2019; Liu et al., 2019) on our task. Significance test is conducted using paired t-test at a significance level of 0.05.

[2] https://nlp.stanford.edu/projects/snli/