

# Discourse-Aware Graph Networks for Textual Logical Reasoning

Yinya Huang, Lemao Liu, Kun Xu, Meng Fang, Liang Lin, *Senior Member, IEEE*,  
Xiaodan Liang, *Senior Member, IEEE*

**Abstract**—Textual logical reasoning, especially question-answering (QA) tasks with logical reasoning, requires awareness of particular logical structures. The passage-level logical relations represent entailment or contradiction between propositional units (e.g., a concluding sentence). However, such structures are unexplored as current QA systems focus on entity-based relations. In this work, we propose logic structural-constraint modeling to solve the logical reasoning QA and introduce discourse-aware graph networks (DAGNs). The networks first construct logic graphs leveraging in-line discourse connectives and generic logic theories, then learn logic representations by end-to-end evolving the logic relations with an edge-reasoning mechanism and updating the graph features. This pipeline is applied to a general encoder, whose fundamental features are joined with the high-level logic features for answer prediction. Experiments on three textual logical reasoning datasets demonstrate the reasonability of the logical structures built in DAGNs and the effectiveness of the learned logic features. Moreover, zero-shot transfer results show the features' generality to unseen logical texts.

**Index Terms**—Natural Language Processing, Logical Reasoning, Question Answering, Multi-Turn Dialogue Reasoning, Graph Neural Networks, Supervised Learning, Zero-shot Learning.



## 1 INTRODUCTION

Natural language understanding in progress is introducing investigation of machines' reasoning capabilities. The recent anticipated logical reasoning requires advanced comprehension of uncovering hidden logical structures. A representative task is logical reasoning QA [2], [3]. It collects questions from standardized exams such as GMAT and LSAT. Each question provides a passage, several answer options, and a question sentence about logical relations, structures, or fallacies. To predict the correct answer, machines need to identify the conclusion and premises in the text and understand how they support or contradict each other. Another representative is multi-turn dialogue reasoning [4], which requires the machine to predict the next utterance that is logically consistent with the conversation.

In principle, logical structures consist of two critical factors, logical components, and logical relations. The core logical components include conclusion and premises, usually complete sentences or subordinate clauses. The logical relations, on the other hand, are mainly entailment, refutation, or contradiction between these sentences. Moreover, the key phrases in the statements indicate

inference patterns. Practically, an example is illustrated in Figure 1. To find the flaw in the argument, one first needs to identify the conclusion and premises. Indicated by the clue words such as “conclude”, “if”, and “then”, the third sentence is the conclusion, whereas the first two sentences provide supporting premises. Indicated by the connectives and the key terms as highlighted, the premises are further decomposed into two entailing structures. From the repeating key terms, one can find the inference patterns  $A \rightarrow B$  and  $\neg A \rightarrow \neg B$  in the two premises, respectively. According to the context, Premise 2 is derived from Premise 1, which then derives the conclusion of  $\neg B$ . However, the reasoning in this argument contradicts the law of contraposition, which is  $A \rightarrow B \vdash \neg B \rightarrow \neg A$ . This leads to the correct option A. In contrast, one can hardly answer this question regardless of the logical structure.

However, many existing deep models often neglect how to mine such appropriate logical structures, and consequently, is hard to learn logic features to handle complex reasoning. For example, traditional deep QA systems [5], [6], [7] and retrieval-based dialogue systems [8], [9] learn to match key entities between the passage and the question. Though mastering previous tasks, they only perform slightly better than random in logical reasoning. More recent QA systems [10], [11], [12], [13] construct discrete structures according to co-occurrence and coreference of named entities and simulate multi-hop reasoning [14], [15] with graph neural networks [16]. Similarly, numerical reasoning systems [17] encode numerical relations between numbers with the topology of graphs. Moreover, current Fact-Checking models [18], [19] and NLI models [20], [21], [22] focus on semantic matching for better knowledge retrieval or estimating the inference type between sentence pairs. In contrast, solving logical reasoning requires awareness of inference patterns beyond knowledge. Therefore, current structures and reasoning processes are insufficient for solving textual logical reasoning, as the core logical structure includes passage-level relations over clause-like units.

- X. Liang is the corresponding author.  
Email address: xdliang328@gmail.com
- Y. H. and X. L. are with the Shenzhen Campus of Sun Yat-sen University, China. L. Lin is with Sun Yat-sen University, China. L. Liu is with Tencent AI Lab. K. X. is with Huawei, and M. F. is with the University of Liverpool.
- This study was done during Y. Huang's internship at Tencent AI Lab.
- Part of this study has been accepted as “DAGN: Discourse-Aware Graph Network for Logical Reasoning” [1] in the Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2021). This paper extends the previous work in the following aspects. First, we add an edge-reasoning mechanism to evolve the constructed logic graphs for adaptive representations, and we conduct experiments in zero-shot scenarios to verify the generalization ability of learned logic representations. We further conduct experiments on dialogue-understanding tasks to investigate the adaptation from formal text to informal language.

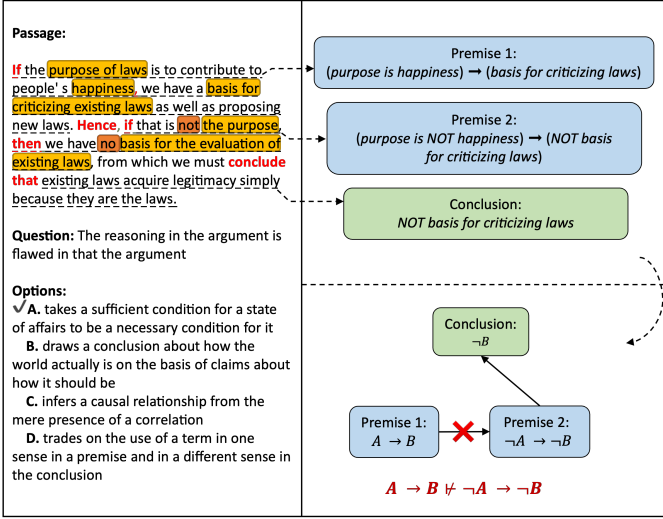


Fig. 1. An example of logical reasoning QA (left) and the logical structure-based solution (right). Inference patterns are found by linguistic clues. The logical units are the conclusion or premises, which are the sentential text spans. The highlighted key terms indicate the logical variables in logical reasoning.

On the other hand, recent advances in transformer-based pre-trained language models (PLMs) [23], [24], [25], [26], [27] have witnessed great success in extensive natural language tasks [28], [29], [30], but fail logical reasoning [2], [3], [4]. The PLMs are trained on large numbers of unlabeled corpora, and the transformer-based architecture with multiple self-attention layers facilitates the encoding of contextualized representations. They learn syntactic and semantic structures in an implicit manner [31]. Besides, several works [21], [22], [32] incorporate explicit syntactic or semantic structural constraints into PLMs and further improve the representations. However, the highlighted token correlations do not guarantee appropriate logical components and relations. Moreover, although the community further observes some reasoning capability [33] from these pure transformer-based models, it is not sufficient for advanced reasoning.

Therefore, several questions are remained open: *How to construct logical structures to benefit the systems for textual logical reasoning? And how to better learn logic representations?*

To this end, we propose discourse-aware graph networks (DAGNs) to focus on inference patterns and learn general logic representations. To do this, DAGNs construct logical structure from the plain text as structural constraints, then learns logic representations by end-to-end evolving the logic relations in the graphs and updating the graph features. Generally speaking, the logic graphs are built via linguistic clues and logic theories so that are easily applied to new text. The logic representation learning applies an edge-reasoning mechanism over the constructed graphs, then conducts graph reasoning to update the logic graph features, which leverages fundamental embeddings from a general encoder such as PLM. Specifically, the logic graph construction uses discourse connectives such as “because” and “if” [34] as text span delimiters. They indicate the logical relations and delimit the texts into clause-like logical units, which is in line with the intuition in informal logic theories [35], [36]. The delimited text spans are regarded as logical reasoning units. The logic graphs are formed with text spans as nodes, connected by linguistic and logical edges.

Logic representation learning is a graph reasoning process. It first discovers advanced logical relations from the constructed logic graphs, for instance, multi-hop relations with different edge types. The relation discovery is an iterative edge selection and propagation procedure inspired by the previous meta-path generation model [37]. Given the updated logical relations, it then initializes the graph features with token embeddings, then performs graph reasoning to aggregate the node embeddings by a node-weighted graph convolutional network. The output multi-hop logic features are further fused with the fundamental embeddings to provide hierarchical features for downstream prediction. The learning process leverages underlying features such as pre-trained contextual embeddings and merely needs a few rounds of fine-tuning, and is therefore efficient.

We conduct comprehensive experiments on three datasets, including two logical reasoning QA datasets [2], [3] and one multi-turn dialogue understanding dataset [4] in both supervised and zero-shot scenarios. In general, DAGNs outperform the state-of-the-art models in supervised settings, showing strong generality in zero-shot transfer. The results show that the edge-reasoning mechanism leads to logical feature generality and model stability. The logic graphs are proved effective for learning general and transferrable logic representations. This indicates the importance of focusing on inference patterns beyond knowledge in logical reasoning tasks.

The contributions of this paper are summarized as follows:

- We explore effective discourse-aware graph networks (DAGNs) for textual logical reasoning. The model constructs logic graphs as structural constraints then learns to identify advanced logical relations and learn logic representations by the graphs.
- The edge-reasoning mechanism evolves the logical relations to adapt the logic representation learning, which results in feature generality and model stability.
- The proposed logic graph construction uses generic textual clues and logic theories and is easily applied to new texts. Meanwhile, graph-based representation learning leverages fundamental encoding techniques; hence is handy for fine-tuning and is widely applicable.
- Experiments on three datasets indicate that DAGNs are superior in textual logical reasoning and provide beneficial logical information. Besides, DAGNs show strong generality to unseen logical questions.

## 2 PRELIMINARIES

### 2.1 Task: Logical Reasoning QA

Logical reasoning QA requires a machine to understand the logic behind the text, for example, identifying the logical components, logical relations, or fallacies.

For multiple-choice logical QA, given a logical passage, a question, and several candidate answer options, a machine needs to predict the answer by understanding the logic of the passage. We give notations for convenient discussion. For a logical reasoning question (passage, question, options), we denote the sequences passage, question, and option as  $S_p, S_q, S_o^c$ , respectively, where  $c \in C$ ,  $c$  is the candidate index and  $C$  is the overall number of candidates. Then a machine’s inputs are  $S^c = [S_p; S_q; S_o^c]$ ,  $c \in C$ , where “;” denotes sequence concatenation.

Similarly, for multi-turn dialogue reasoning, a machine is given dialogue context and multiple candidate responses and is required to

give the logically correct response according to the dialogue context. For a single dialogue (dialogue context, candidate responses), we denote the sequences dialogue context and candidate response as  $S_d$  and  $S_r$ , respectively. The machine’s inputs are  $S^c = [S_d; S_r^c]$  for each  $c \in C$ . Predicting the answer from  $C$  options needs to give ranking scores  $p^c$  for all  $c \in C$ .

## 2.2 Logic Theories for Logical Reasoning QA

Logic theories study symbolic reasoning processes in daily language use. It can be generally grouped into informal logic [35], [36] and formal logic [38]. The informal logic uncovers reasoning structure in context. In contrast, formal logic extracts the language into symbolic axiomatic systems to evaluate its validity. Both inspire the modeling for logical reasoning QA.

### 2.2.1 Informal Logic

**Logical Components in Arguments.** Informal logic [35], [36] studies the structural reasoning processes in argumentation. The structure is named argument [39]. An example argument is:

*A and B; therefore C.*

Here, “A”, “B” and “C” are propositions, and “C” is a conclusion drawn from the two premises “A” and “B”. Hence in this discrete structure, conclusion and premise are two fundamental logical components, which are usually complete sentences or sub-sentences [40].

**Inference Indicators.** To uncover the logical components from text and reconstruct the structure, informal logic has organized frequently encountered indicators that prompt the premise or conclusion. Representative premise indicators involve “since”, “because”, “for”, “given that” and so forth. Meanwhile, conclusion indicators include “therefore”, “so”, “consequently” and others.

Inspired by these, we reconstruct logical structures for logical reasoning QA by leveraging such inference indicators as text delimiters, which segment the passage into multiple sentences or clauses that properly are the basic reasoning units. The indicators themselves then signify corresponding logical relations between the units.

### 2.2.2 Formal Logic

**Deviation of Logical Expressions.** In formal logic system such as first-order logic (FOL), extensive well-formed formulae (i.e., logical expressions) are derived from a few axioms and rules. The soundness of derivation guarantees that the derived expressions are true if only the axioms are true [38].

For example, in first-order propositional logic, the modus ponens rule is as follows:

$$P \rightarrow Q, P \vdash Q. \quad (1)$$

Thus, if  $\alpha \wedge \beta \rightarrow \gamma$  is an axiom and is true, and  $\alpha \wedge \beta$  is true, then it is derived that  $\gamma$  is true.

Another example is that given that we have the rule of addition:

$$P \vdash P \vee Q, \quad (2)$$

then say  $\alpha \rightarrow \beta$  is an axiom and is true, then  $(\alpha \rightarrow \beta) \vee \gamma$  as a derived expression is true.

Therefore, it is observed that in the logical expression derivation, the expressions that are derived from each other are correlated

only if they have shared variables, such as the  $\alpha \wedge \beta$  in the first example and the  $\alpha \rightarrow \beta$  in the second one. This motivates us to build the variable edges in the logic graph construction.

**Validity of Expressions and Instantiation.** If a logical expression is valid, its multiple instantiations are true as they follow the same valid reasoning process. For instance, two instantiations of the modus ponens rule in eq. (1) are as follows:

**Example 2.1** (Instantiation of modus ponens). “All men are mortal. Socrates is a man. Therefore, Socrates is mortal.” It is obtained by grounding  $P$  to “be\_men”, and  $Q$  to “be\_mortal”.

**Example 2.2** (Instantiation of modus ponens). “All birds can fly. Eagles are birds. Therefore, eagles can fly.”. It is obtained by grounding  $P$  to “be\_bird”,  $Q$  to “can\_fly”.

We can tell that the statements in Example 2.1 and Example 2.2 are true. Albeit they are in diverse topics, as we know that their shared reasoning skeleton, i.e., the modus ponens rule, is valid.

Furthermore, in logical texts, the logical reasoning processes are performed in a natural language format. The logical variables are embedded. One of the hints for such logical variables is the topic-related terms, which are mainly the recurring topic words or phrases, such as the “men” and “mortal” in Example 2.1 and the “birds” and “fly” in Example 2.2. Accordingly, we provide topic-related terms detection in our graph node construction.

## 3 DISCOURSE-AWARE GRAPH NETWORKS

The proposed discourse-aware graph networks (DAGNs) have two main components: logic graph construction and logic representation learning. The logic graph construction contains strategies of logical unit delimitation, topic-related term detection, graph node arrangement, and graph edge definition. Meanwhile, logic representation learning is a graph reasoning process that takes contextual encoding as input, updates features with the logic graph constraints, merges multiple features, and is trained end-to-end for logical QA prediction.

§3.1 introduces the overall strategy of logic graph construction. §3.2 describes the logic representation learning process. The overlook of DAGNs is demonstrated in Figure 2.

### 3.1 Logic Graph Construction

Given a logical reasoning question (passage, question, options) or (dialogue context, candidate responses), which is formalized as  $S^c, c \in C$  as described in §2.1, we construct logic graphs  $\mathcal{G}^c = \{\mathcal{V}^c, \mathcal{E}^c\}, c \in C$ .

We describe the graph node and edge definition separately. The graph nodes are text’s segmented sentences or sub-sentences, indicated by discourse-aware connectives. Each node is further attached with topic-related terms and is assigned a node type. As for the graph edges, discourse-connective edges and variable edges link the nodes differently. The overall construction is illustrated in Figure 3.

#### 3.1.1 Nodes via Discourse Unit Delimitation

It is studied that clause-like text spans delimited by discourse relations can be discourse units that reveal the rhetorical structure of texts [34], [41]. We further observe that such discourse units are essential logical propositions in logical reasoning, such as premise or conclusion. As the example shown in Figure 3, the “while” in the passage indicates a comparison between the attributes of the



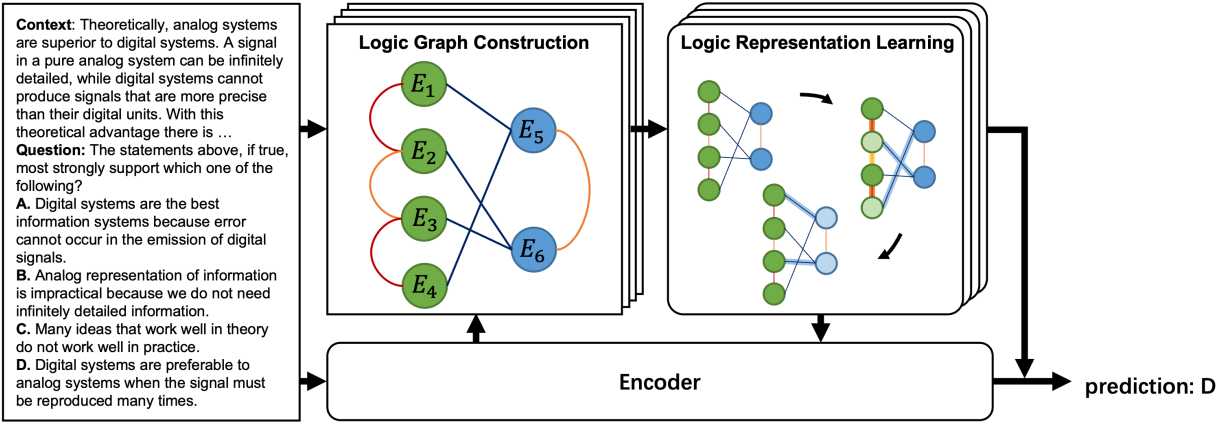


Fig. 2. The discourse-aware graph networks (DAGNs) pipeline mainly consists of (1) logic graph construction (2) logic representation learning. The logic graph construction module takes a logical QA data point as input and constructs logic graphs. The logic representation learning module then performs graph reasoning upon the constructed logic graphs. Besides, the encoder provides fundamental embeddings for the pipeline.

“analog system” and that of the “digital system”. The “because” in the option uncovers that “error cannot occur in the emission of digital signals” as a premise to the conclusion “digital systems are the best information systems”.

This observation is agreed with informal logic theories [35], [36], which study uncovering logical structure from the texts and have conventional in-line logical indicators. For example, acknowledged premise indicators include “since”, “because”, “given that”. Conclusion indicators include “therefore”, “so”, “consequently”, and so forth. Most of these indicators are discourse connectives.

Some discourse parsers [42], [43] perform discourse unit segmentation. However, discourse parsing is still challenging, and the parsers are not general to new data, such as logical reasoning questions. For example, SegBot [43] is good on the RST-DT dataset but does not work well on the standardized exam texts as in the ReClor dataset. Thus, we customize discourse unit delimitation strategy for logical texts.

We use the Penn Discourse TreeBank (PDTB 2.0) [34] to help draw discourse connectives. PDTB 2.0 contains discourse relations that are manually annotated on the 1 million Wall Street Journal (WSJ) corpus and are broadly characterized into “Explicit” and “Implicit” connectives. The former ones are explicitly present in sentences such as discourse adverbial “instead” or subordinating conjunction “because”, whereas the latter ones are inferred by PDTB annotators between successive pairs of text spans split by punctuation marks such as “.” or “;”. We take all the “Explicit” connectives as well as common punctuation marks to form our discourse-aware delimiter library, presented in Table 1. Each logical text is split into elementary discourse units (EDUs) by all the delimiters in the library. The EDUs are taken as graph nodes  $\mathcal{V}$ .

**Nodes with Topic-Related Terms.** The desired key terms are those real nouns or phrases that repeatedly appear in the text. Such nouns or phrases are instantiations of logical variables in propositions. As a result, replacing such terms with abstract variables or terms in other topics does not change the process of reasoning. For example, in Figure 3, the first two sentences indicate a comparison of “signal” between “analog system(s)” and “digital system(s)”. Performing abstraction by replacing “signal” with variable  $\gamma$ , “analog system(s)” with variable  $\alpha$ , and “digital system(s)” with variable  $\beta$ , the propositions are free from the topic of electronics, but the comparison relation is retained.

We use a sliding window to collect the recurring phrases. Given the input logical text, stemming is first applied to handle morphological diversity. Then, the sliding window loops over n-grams and records the reoccurrence. Next, all the stop words and overlapped substrings are filtered. The resulting topic-related terms are attached to the nodes according to which text segment they belong.

**Binary Node Types.** The text of logical reasoning QA consists of two possible structures: (passage, question, options) or (dialogue context, candidate responses). We regard passage or dialogue context as context texts that carry the main logical reasoning structure, whereas regard (question, options) or candidate responses as candidate texts that are added to the context texts and should remain their logical consistency.

According to the discourse unit delimitation, the graph nodes are naturally from the context texts or the candidate texts. Therefore, we define two disjoint and independent node sets: context node set  $\mathcal{V}_u$  and candidate node set  $\mathcal{V}_v$ .  $\mathcal{V}_u \cup \mathcal{V}_v = \mathcal{V}$  and  $\mathcal{V}_u \cap \mathcal{V}_v = \emptyset$ . The interplay between the two node sets formulates logical consistency between the context and the candidate texts.

### 3.1.2 Edge Definition

**Discourse-Connective Edges.** We directly use the discourse-aware delimiters to build the discourse-connective edges. The intuition is that the delimiters indicate the in-line logical relations, as demonstrated in informal logic theories [35], [36]. Therefore, the “Explicit” connectives and the punctuation marks are taken as two types of edges, and we name them *explicit-connective edges* and *implicit-connective edges*, respectively. One edge is added between the EDUs before and after each delimiter, with the edge type corresponding to the delimiter. If “Explicit” and “Implicit” connectives are present simultaneously, we choose only to use the “Explicit” connectives. Besides, considering the disjoint node sets  $\mathcal{V}_u$  and  $\mathcal{V}_v$ , the discourse-connective edges only connect nodes within the same node-set. The edges are undirected.

As shown in Figure 3, the two nodes  $\text{EDU}_2 = \text{“digital systems cannot produce signals that ... units”}$  and  $\text{EDU}_3 = \text{“With ... disadvantage”}$  are connected with an implicit-connective edge. The nodes  $\text{EDU}_1 = \text{“A signal in a pure analog system ... detailed”}$  and  $\text{EDU}_2 = \text{“digital systems cannot produce signals that ... units”}$



TABLE 1  
The discourse-aware delimiter library.

<b>Explicit Connectives</b>	once, although, though, but, because, nevertheless, before, for example, until, if, previously, when, and, so, then, while, as long as, however, also, after, separately, still, so that, or, moreover, in addition, instead, on the other hand, as, for instance, nonetheless, unless, meanwhile, yet, since, rather, in fact, indeed, later, ultimately, as a result, either or, therefore, in turn, thus, in particular, further, afterward, next, similarly, besides, if and when, nor, alternatively, whereas, overall, by comparison, till, in contrast, finally, otherwise, as if, thereby, now that, before and after, additionally, meantime, by contrast, if then, likewise, in the end, regardless, thereafter, earlier, in other words, as soon as, except, in short, neither nor, furthermore, lest, as though, specifically, conversely, consequently, as well, much as, plus, and, hence, by then, accordingly, on the contrary, simultaneously, for, in sum, when and if, insofar as, else, as an alternative, on the one hand on the other hand
<b>Punctuation Marks (Implicit Connectives)</b>	. , ; : ? ! <s> </s>

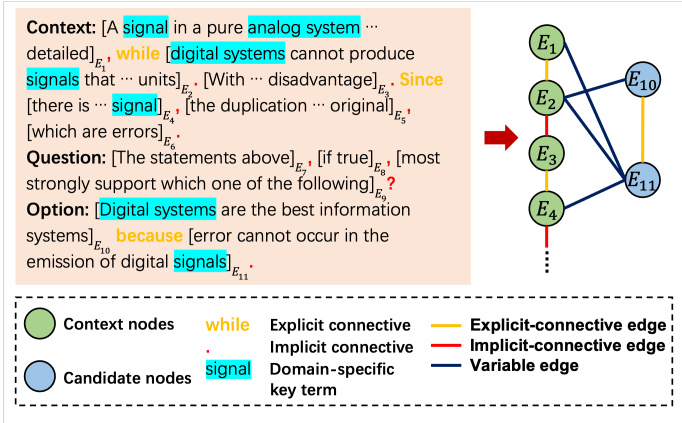


Fig. 3. The logic graph construction is based on in-line discourse connectives which split the text into segments as logical units and form the graph nodes.

are joint with the explicit-connective edge when both “,” and “while” are between them. Besides, the nodes EDU<sub>6</sub> and EDU<sub>7</sub> are adjacent in the input text, but there is no discourse-connective edge between them because they are from different node sets.

As a comparison, we also try different edge linking strategies for the discourse-connective edges, including random edge linking, full-connection, and single edge type. We further discuss these strategies and their benefits to logical reasoning in Section 4.5.1.

Given the binary node sets  $\mathcal{V}_u$  and  $\mathcal{V}_v$ , we denote the adjacency matrices of explicit-connective and implicit-connective edges as:

$$A^E = \begin{pmatrix} A_u^E & 0_{u,v} \\ 0_{v,u} & A_v^E \end{pmatrix} \quad \text{and} \quad A^I = \begin{pmatrix} A_u^I & 0_{u,v} \\ 0_{v,u} & A_v^I \end{pmatrix},$$

where  $A_*^E$  and  $A_*^I$  denote the inner-set edge linkings.

**Variable Edges.** Variable edges connect the disjoint context nodes  $\mathcal{V}_u$  and candidate nodes  $\mathcal{V}_v$ , representing the derivations between logical propositions. The intuition is that when the candidate nodes from the correct option are joined with the context nodes, the logical consistency is retained, while the intervention of the candidate nodes from the distracting options will disturb the logic graphs.

For simulating such logical consistency as in logical expression derivation, edges are added to those EDU nodes that carry at least one shared variable. Practically, the variables are regarded as the tagged topic-related terms. Thus, given the disjoint node sets, if a

node pair shares a topic-related term, an edge is added between them.

As illustrated in Figure 3, EDU<sub>2</sub> = “digital systems cannot produce signals that ... units” and EDU<sub>10</sub> = “digital systems are the best information systems” represent two propositions, and they share the key term “digital systems”, therefore they are connected with a variable edge. Similarly, EDU<sub>1</sub> = “a signal ... detailed” and EDU<sub>11</sub> = “error cannot occur in the emission of digital signals” share the key term “signal” and are connected with a variable edge. The edges are undirected.

Formally, given the binary node sets  $\mathcal{V}_u$  and  $\mathcal{V}_v$ , for each node pair  $(v_u, v_v)$ , where  $v_u \in \mathcal{V}_u$  and  $v_v \in \mathcal{V}_v$ , when there is a key term  $\kappa$  that  $\kappa \in v_u$  and  $\kappa \in v_v$ , a variable edge is added between them. As a result, the adjacency matrix of the variable edges is:

$$A^S = \begin{pmatrix} 0_u & B_{u,v}^S \\ B_{v,u}^S & 0_v \end{pmatrix},$$

where  $B_{u,v}^S$  and  $B_{v,u}^S$  are incidence matrices between  $\mathcal{V}_u$  and  $\mathcal{V}_v$ .

### 3.2 Logic Representation Learning

Given a logical question and its constructed graphs, we now build a logic-based model that is end-to-end trained for logic representation learning. The model takes the question and graphs as input, encodes the input sequence, conducts edge evolving and graph reasoning to produce logic representations, then fuses the fundamental encodings for downstream prediction. The reasoning module is a plugin module to a general encoder and leverages the contextual features. Hence the overall model only needs a few rounds of fine-tuning for feature updates. Figure 4 demonstrates the learning pipeline.

#### 3.2.1 The End-to-End Learning Pipeline

**Text Inputs.** For a logical question, the input sequences  $S^c$ ,  $c \in C$  are formulated as described in Section 2.1. Each  $S^c$  is further truncated into tokens  $S^c = (s_1^c, s_2^c, \dots, s_L^c)$  where  $L$  denotes the number of tokens.

**Graph Inputs.** Each  $S^c$  has a corresponding logic graph  $\mathcal{G}^c$ . The nodes correspond to elementary discourse units (EDUs) in  $S^c$ , which are recorded by  $D^c(l) = n$ , a position mapping from token position  $l$  to segment position  $n$ .  $n \leq N$ ,  $l \leq L$  with  $L$  tokens and  $N$  EDUs in total. The edges are of three types, and the model takes their adjacency matrices  $\{A^{c,E}, A^{c,I}, A^{c,S}\}$ .

**Token Encoding.** The  $S^c$ ,  $c \in C$  are individually fed into a shared encoder  $E$  and obtain the token embeddings:  $E(S^c) =$

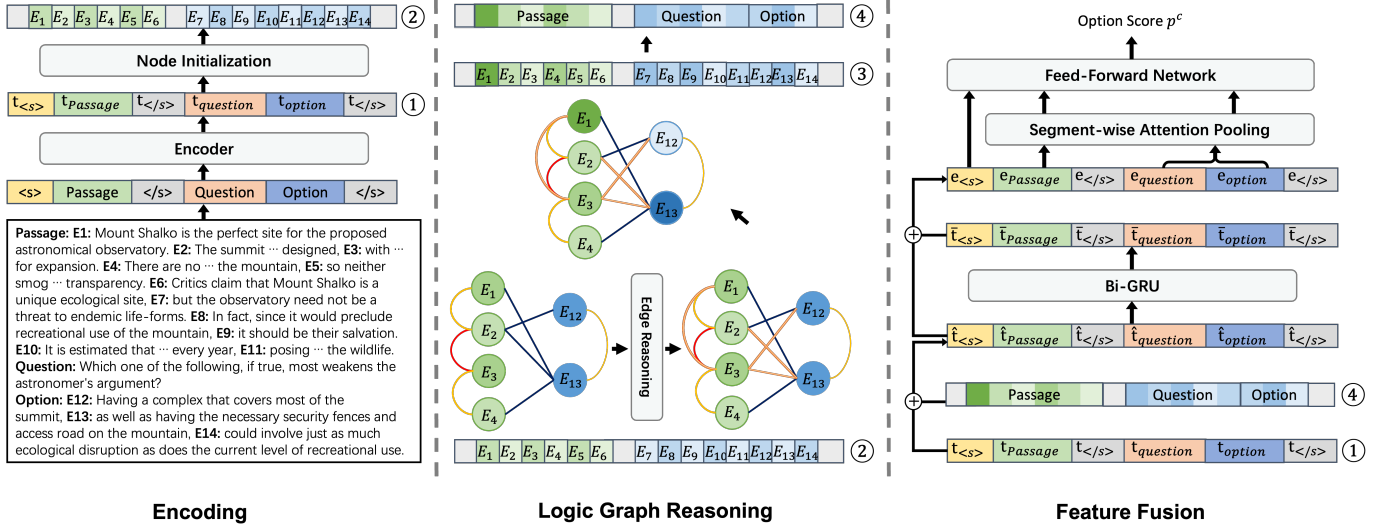


Fig. 4. The logic representation learning process. Logic graph reasoning starts with node initialization from an encoder and produces logic representations. The initial token embeddings and the high-level logic embeddings are used for downstream prediction.

( $\mathbf{t}_1^c, \mathbf{t}_2^c, \dots, \mathbf{t}_L^c$ ), where  $\mathbf{t}_*^c \in \mathbb{R}^b$  and  $b$  is the dimension of a token embedding.

**Logic Edge Reasoning.** Given the adjacency matrices  $\{A^{c,E}, A^{c,I}, A^{c,S}\}$ , a module softly selects the edge types, then perform matrix multiplication to propagate new edges. The soft propagated edges are then converted into adjacency matrices  $\{A^{c,(h)}\}_{h \in H}$ , and  $H$  is the maximum hops of graph reasoning. The set of adjacency matrices are then updated with the propagated edges  $\bar{A} = \{A^{c,E}, A^{c,I}, A^{c,S}\} \cup \{A^{c,(h)}\}_{h \in H}$ . As a result, the evolved graph  $\bar{\mathcal{G}}^c$  contains the multi-hop inference edges derived from hybrid logical relations. The parameters in the soft edge selection are updated via end-to-end training.

**Logic Graph Reasoning.** Given the token embeddings ( $\mathbf{t}_1^c, \mathbf{t}_2^c, \dots, \mathbf{t}_L^c$ ) and the graph inputs  $D^c(l) = n, \{A^{c,E}, A^{c,I}, A^{c,S}\} \cup \{A^{c,(h)}\}_{h \in H}$ , logic representations are learned via graph reasoning. Node embeddings are initialized by merging the token embeddings according to  $D^c(\cdot)$ , then are updated via multi-step message propagation through the adjacency matrices. Afterward, the updated node embeddings are assigned to each token by  $D^c(\cdot)$  again as the learned logic representation for each token.

**Feature Fusion.** For each token, the learned high-level logic representation and the fundamental contextual embedding are fused. Furthermore, the token embeddings are pooled for downstream prediction. For each option  $c$ , the model obtains a pooled embedding  $\hat{\mathbf{p}}^c$ .

**Option Ranking.** Each option embedding  $\hat{\mathbf{p}}^c$  is fed into a linear layer to get a ranking score. Furthermore, the probabilities for selecting the options are obtained by a softmax function:

$$\hat{p}^c = \mathbf{W}\hat{\mathbf{p}}^c + \mathbf{b}, \quad (3)$$

$$p^c = \frac{e^{\hat{p}^c}}{\sum_{c \in C} e^{\hat{p}^c}}. \quad (4)$$

**Overall Objective Function.** Given single question input (passage, question, options) or (dialogue context, candidate responses), the model is end-to-

end trained by cross-entropy loss with option labels  $y^c$ :

$$\mathcal{L} = - \sum_{c \in C} y^c \log(p^c). \quad (5)$$

### 3.2.2 Logic Edge Reasoning

The edge-reasoning mechanism is demonstrated in Algorithm 1. Given a logic graph with three edge types, we concatenate their corresponding adjacency matrices with an identity matrix  $\bar{A}^{(0)} = [A^E; A^I; A^S; I]$ . The soft edge selection weighted sum the adjacency matrices  $\bar{A}^{(0)}$ , and outputs the soft selected edges  $\Gamma^{(0)}$ :

$$\Gamma^{(0)} = \bar{A}^{(0)} \cdot \text{softmax}(\mathcal{W}^{(0)}). \quad (6)$$

where  $\mathcal{W}^{(0)} \in \mathbb{R}^{N \times N}$  is a weight matrix initialized with normal distribution.

Then the edge reasoning is performed in an iterative manner and updates the final edge set  $\bar{A}$ . During the process, another soft edge selection is performed and yields  $\hat{\Gamma}$ . Then given the  $\hat{\Gamma}$  and the soft edge matrix from the last iteration  $\Gamma^{(i-1)}$ , an edge propagation is performed by matrix multiplication between them, and produces the  $i$ -hop soft edge matrix:

$$\Gamma^{(i)} = \Gamma^{(i-1)} \hat{\Gamma}. \quad (7)$$

The resulting  $\Gamma^{(i)}$  is converted into a new adjacency matrix  $\bar{A}^{(i)}$  if the soft edge element exceeds a threshold  $\delta$ , which is added to the final edge set  $\bar{A}$ .

To further increase the diversity of hybrid edges, the edge reasoning process is repeated for  $d$  times and also updates  $\bar{A}$ . The logic graph is then updated with the hybrid edges  $\bar{\mathcal{G}} = (\mathcal{V}, \mathcal{E} \cup \mathcal{E}^H)$ , where  $\mathcal{E}^H$  is the edge set corresponds to  $\bar{A}$ .

### 3.2.3 Logic Graph Reasoning

This section illustrates the detailed logic representation learning process. This process is conducted via graph reasoning by a graph neural network. It consists of node initialization, graph reasoning, and logic embedding assignment for each token.

**Node Initialization.** The graph nodes are EDUs, therefore they are initialized with token embeddings to leverage contextual information. Given token embeddings ( $\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_L$ ) and

---

**Algorithm 1:** Logic Edge Reasoning

---

**Input:** A logic graph  $\mathcal{G}$  with explicit-connective edges  $A^E$ , implicit-connective edges  $A^I$ , and variable edges  $A^S$ , identity matrix  $I$ , edge-extraction threshold  $\delta$ , the max hop  $H$

**Output:** The set of hybrid logical edges  $\bar{A}$

```

1 // Initialization
2  $\bar{A}^{(0)} \leftarrow [A^E; A^I; A^S; I]$ ,  $\mathcal{W}^{(0)} \leftarrow \mathcal{N}(0, 1)$ 
3  $\Gamma^{(0)} \leftarrow \text{edgeSelection}(\bar{A}^{(0)}, \mathcal{W}^{(0)})$ 
4  $\bar{A} \leftarrow \emptyset$ 
5 // Edge Reasoning
6 for  $i = 1 : H$  do
7    $\mathcal{W}^{(i)} \leftarrow \mathcal{N}(0, 1)$ 
8    $\hat{\Gamma} \leftarrow \text{edgeSelection}(\bar{A}^{(i-1)}, \mathcal{W}^{(i)})$ 
9    $\Gamma^{(i)} \leftarrow \text{edgePropagation}(\Gamma^{(i-1)}, \hat{\Gamma})$ 
10   $\bar{A}^{(i)} \leftarrow \text{edgeExtraction}(\Gamma^{(i)}, \delta)$ 
11   $\bar{A} \leftarrow \bar{A} \cup \{\bar{A}^{(i)}\}$ 
12 end
13 return  $\bar{A}$ 

```

---

the logical unit delimitations  $D(l) = n$ , the node embedding corresponding to the  $n$ -th EDU (denoted as  $U_n$ ) is then calculated via  $\mathbf{v}_n^{(0)} = M(\bigwedge_{D(l) \in U_n} \mathbf{t}_l)$ , where  $D(l) \in U_n$  denotes the tokens in the  $n$ -th EDU and  $M$  is the merging function. Specifically, we use a trivial  $M$ , which is sum pooling the token embeddings:  $\mathbf{v}_n^{(0)} = \sum_{D(l) \in U_n} \mathbf{t}_l$ .

**Graph Reasoning.** Given a logic graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , where  $\mathcal{E} = \mathcal{E}^E \cup \mathcal{E}^I \cup \mathcal{E}^S \cup \mathcal{E}^H$ , for a node  $v_i \in \mathcal{V}$ ,  $\mathcal{N}_i = \{j | (v_j, v_i) \in \mathcal{E}\}$  indicates its neighbors. The node embeddings are updated via: the explicit-connective edges  $\mathcal{E}^E$ , the implicit-connective edges  $\mathcal{E}^I$ , the variable edges  $\mathcal{E}^S$ , and the hybrid edges  $\mathcal{E}^H$ . The corresponding adjacency matrices are  $A^E, A^I, A^S$  and  $\bar{A}$ .

For stability, we first normalize the variable matrix  $A^S = \begin{pmatrix} 0_u & B_{u,v}^S \\ B_{v,u}^S & 0_v \end{pmatrix}$  with:

$$\hat{B}_{u,v}^S = D_{u,v}^{-1} B_{u,v}^S, \quad \hat{B}_{v,u}^S = D_{v,u}^{-1} B_{v,u}^S \quad (8)$$

where  $D_{u,v}^{-1}$  is the degree matrix of  $B_{u,v}^S$  and similar to  $D_{v,u}^{-1}$ .

Then node features are updated via multiple graph learning layers to obtain multi-hop logic representations. For node  $v_i$ , its initial node embedding is  $\mathbf{v}_i^{(0)}$ . Given node embedding  $\mathbf{v}_i^{(k-1)}$  from the  $(k-1)$ -th layer, a node weight is first calculated via linear transformation with a sigmoid function  $\sigma$ :

$$\alpha_i = \sigma(\mathbf{W}^\alpha (\mathbf{v}_i^{(k-1)}) + \mathbf{b}^\alpha), \quad (9)$$

then message propagation is conducted by simultaneously considering three relation types and taking information from the neighbors  $\mathbf{v}_j \in \mathcal{N}_i$ :

$$\tilde{\mathbf{v}}_j^{(k-1)} = \mathbf{W}^\gamma \mathbf{v}_j^{(k-1)} + \mathbf{b}^\gamma, \quad (10)$$

$$\tilde{\mathbf{v}}_i^{(k-1)} = \frac{1}{|\mathcal{N}_i|} \left( \sum_{j \in \mathcal{N}_i} \sum_{E \in \{E, I, S\}} \alpha_j A_{ji}^E \tilde{\mathbf{v}}_j^{(k-1)} \right). \quad (11)$$

The node embedding for the  $k$ -th layer is finished by joining the embeddings:

$$\mathbf{v}_i^{(k)} = \text{ReLU}(\mathbf{W}^\eta \mathbf{v}_i^{(k-1)} + \tilde{\mathbf{v}}_i^{(k-1)} + \mathbf{b}^\eta), \quad (12)$$

where  $\mathbf{W}^\eta$  and  $\mathbf{b}^\eta$  are weight and bias respectively.

**Global Graph Representation.** The updated node embeddings  $\{\mathbf{v}_i\}_{i \in N}$  are fed into a dot-product self-attention layer [44] and obtains  $\{\mathbf{v}_i^G\}_{i \in N}$ , which are then weighted summed into the global graph representation. The weights  $\alpha_i^G$  are simply set to 1 in this case.

$$\mathbf{v}_G = \sum_{i \in N} \alpha_i^G \mathbf{v}_i^G. \quad (13)$$

**Token-wise Logical Embeddings.** The updated node embeddings are assigned to each token. For each  $l \in L$ , based on  $D(l) = n$ , we have:

$$\mathbf{t}_l^\lambda = \mathbf{v}_n. \quad (14)$$

### 3.2.4 Feature Fusion

After logic representation learning, each token, now has an original token embedding, and a logical embedding. The start-token embedding pairs to the global graph representation  $\mathbf{v}_G$ , representing the correspondence between the text and the structure. The embeddings are fused with a hierarchical fusion, followed by pooling.

**Hierarchical Fusion.** For each token  $s_l^c \in S^c$ ,  $l \leq L$ , the fundamental token embeddings  $\mathbf{t}_l$  and the high-level logic embeddings  $\mathbf{t}_l^\lambda$  are added up, followed by a layer normalization [45]:

$$\hat{\mathbf{t}}_l = \text{LayerNorm}(\mathbf{t}_l^\lambda + \mathbf{t}_l). \quad (15)$$

The resulting token embedding sequence  $(\hat{\mathbf{t}}_0, \hat{\mathbf{t}}_1, \hat{\mathbf{t}}_2, \dots, \hat{\mathbf{t}}_L)$  are further fed into a bidirectional GRU [46] with residual structure [47] and layer normalization:

$$\bar{\mathbf{t}}_l = \text{Bi-GRU}(\hat{\mathbf{t}}_l), \quad (16)$$

$$\mathbf{e}_l = \text{LayerNorm}(\hat{\mathbf{t}}_l + \bar{\mathbf{t}}_l). \quad (17)$$

**Segment-wise Pooling.** The hierarchically fused embeddings  $(\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_L)$  are separated into three segments: the first-token segment  $\mathbf{e}_1$ , the passage segment  $\{\mathbf{e}_*^p\} = (\mathbf{e}_2, \dots, \mathbf{e}_M)$ , and question-option segment  $\{\mathbf{e}_*^o\} = (\mathbf{e}_{M+1}, \dots, \mathbf{e}_L)$ ,  $1 < M < L$ . The passage embeddings and the question-option embeddings are further merged into two single embeddings  $\mathbf{e}^p$  and  $\mathbf{e}^o$  via segment-wise attention pooling, respectively:

$$\alpha_p = \frac{e^{e_m^p}}{\sum_{m \in [2, M]} e^{e_m^p}}, \quad \mathbf{e}^p = \sum_{m \in [2, M]} \alpha_p \mathbf{e}_m^p, \quad (18)$$

$$\alpha_o = \frac{e^{e_m^o}}{\sum_{m \in [M+1, L]} e^{e_m^o}}, \quad \mathbf{e}^o = \sum_{m \in [M+1, L]} \alpha_o \mathbf{e}_m^o. \quad (19)$$

At last, the three segment-wise embeddings are integrated via concatenation and a single-layer perceptron with normalization:

$$\mathbf{e} = [\mathbf{e}_1; \mathbf{e}^p; \mathbf{e}^o], \quad (20)$$

$$\hat{\mathbf{p}} = \text{LayerNorm}(\text{GeLU}(\mathbf{W}^\sigma \mathbf{e} + \mathbf{b}^\sigma)). \quad (21)$$

## 4 EXPERIMENT

To validate the logic graph construction and the representation learning, we conduct experiments on three textual logical reasoning datasets, including logical reasoning QA and multi-turn dialogue reasoning. We analyze and discuss graph construction and model components in representation learning. Besides, we conduct a generalization test among the datasets via zero-shot learning.



TABLE 2

Experimental results on ReClor dataset. Accuracies (%) are reported. Test-E and Test-H represent the EASY and HARD set of ReClor testing, respectively.

Method	ReClor			
	Dev	Test	Test-E	Test-H
Chance	25.00	25.00	25.00	25.00
Human	-	63.00	57.10	67.20
Ceiling Performance	-	100.00	100.00	100.00
<b>Semantic Matching</b>				
FastText [48]	32.00	30.80	40.20	23.40
Bi-LSTM	27.80	27.00	26.40	27.50
<b>Transformer-based PLMs</b>				
GPT [23]	47.60	45.40	73.00	23.80
GPT-2 [49]	52.60	47.20	73.00	27.00
BERT-Large-MC [24]	53.80	49.80	72.00	32.30
RoBERTa-Large-MC [25], [50]	62.60	55.60	75.50	40.00
<b>Graph Models</b>				
Focal Reasoner (RoBERTa <sub>Large</sub> ) [51]	66.80	58.90	77.05	44.64
DAGNs (RoBERTa <sub>Large</sub> )	66.80	61.00	79.09	46.79
<b>Data-Augmented Methods</b>				
LReasoner (RoBERTa <sub>Large</sub> ) [52]	66.20	62.40	81.40	47.50
LReasoner (RoBERTa <sub>Large</sub> ) <sup>†</sup>	64.70	58.30	77.60	43.10
MERIt (RoBERTa <sub>Large</sub> ) [53]	67.80	60.70	79.60	45.99
DAGNs + LReasoner (RoBERTa <sub>Large</sub> )	69.00	61.90	79.55	48.04
DAGNs + MERIt (RoBERTa <sub>Large</sub> )	68.40	62.40	80.45	48.21

<sup>†</sup> This result is reproduced and reported by MERIt [53].

## 4.1 Datasets

ReClor [2] is a multiple-choice QA dataset with 6,138 logical reasoning questions modified from standardized tests such as GMAT and LSAT. The questions are split into train/dev/test sets with 4,638/500/1,000 questions respectively. ReClor contains 17 question types, including questions about logical components (such as “Necessary Assumptions”, “Sufficient Assumptions”), logical relations (such as “Strengthen”, “Weaken”), reasoning evaluation (such as “Evaluation”, “Technique”) and so forth. The passages contain a mass of complex sentences with uncommon words. The training set and the development set are available. The test set is hold-out and split into an EASY subset and a HARD subset according to the performance of the BERT-base model [24]. The test results are obtained by submitting the test predictions to the leaderboard. The evaluation metric is accuracy.

LogiQA [3] is also a multiple-choice QA dataset with logical reasoning questions. It consists of 8,678 questions collected from the National Civil Servants Examinations of China and manually translated into English by professionals. LogiQA contains 5 question types. It shares some of the reasoning types with ReClor, for example, “Sufficient Conditional Reasoning”. The texts are less lexically complex than that in ReClor. The dataset is randomly split into train/dev/test sets with 7,376 / 651 / 651 samples respectively.

MuTual [4] is a multi-turn dialogue reasoning dataset that evaluates logical reasoning in retrieval-based dialogue systems. The response selection task has four candidate responses for each dialogue, all relevant to the dialogue context, but only one is logically correct. The distracting answers are highly lexically overlapped with the context; hence it is challenging to solve text matching solely. The modified version MuTual<sup>plus</sup> includes a safe response (e.g., “Could you repeat that?”) among the candidates and is more challenging in logical reasoning. The evaluation metrics include recall at position 1 (R@1), recall at position 2 (R@2), and

TABLE 3

Experimental results on LogiQA dataset. Accuracies (%) are reported.

Method	LogiQA	
	Dev	Test
Chance	25.00	25.00
Human	-	86.00
Ceiling	-	95.00
<b>Lexical Matching</b>		
Word Matching [54]	27.49	28.37
Sliding Window [55]	23.58	22.51
<b>Deep QA Systems</b>		
Stanford Attentive Reader [5]	29.65	28.76
Gated-Attention Reader [6]	28.30	28.98
Co-Matching Network [6]	33.90	31.10
<b>Transformer-based PLMs</b>		
BERT-Large-MC [24]	34.10	31.03
RoBERTa-Large-MC [25], [50]	35.02	35.33
<b>Graph Models</b>		
Focal Reasoner (RoBERTa <sub>Large</sub> ) [51]	41.01	40.25
DAGNs (RoBERTa <sub>Large</sub> )	39.63	42.09
<b>Data-Augmented Methods</b>		
LReasoner (RoBERTa <sub>Large</sub> ) <sup>‡</sup> [52]	36.10	38.86
MERIt (RoBERTa <sub>Large</sub> ) [53]	42.40	41.50
DAGNs + LReasoner (RoBERTa <sub>Large</sub> )	40.86	42.24

<sup>‡</sup> We applied the official code on the LogiQA data.

Mean Reciprocal Rank (MRR) in 4 candidate responses. Since the passages are dialogues between two speakers, this dataset has more verbal and informal texts than ReClor and LogiQA. The dataset is randomly split into training, development, and test sets with an 8:1:1 ratio.

## 4.2 Implementation Details

**Logic Graph Construction.** For multiple-choice QA (ReClor and LogiQA), each question contains a passage, a question, and several candidate options. Similarly, each sample contains a dialogue context and multiple candidate responses in the dialogue reasoning dataset. Therefore, considering the different contexts of the candidates, we construct logic graphs for each candidate by pairing every candidate with the passage and the question.

**Graph Reasoning.** For ReClor and LogiQA, we set the maximum length of the input token sequence to 256. The input format is “<s> passage </s> question || option </s>”, where <s> and </s> are the special tokens for RoBERTa [25] model, and || denotes concatenation, following previous works [2], [3]. And the number of stacked GNN layers is 2 for ReClor and 3 for LogiQA. The model is optimized with AdamW [58] with the learning rates 1e-5 for graph reasoning and 5e-6 for parameters. The epsilon is set to 1e-6. A linear scheduler is used and the warmup steps are set to 4,000.

For MuTual, the maximum input length is set to 320. For the dialogue context sequence, we insert a separator token (“</s>” for RoBERTa and “[SEP]” for ELECTRA) between each adjacent utterance pair, following [57]. And the GNN iteration step is 1. The model is optimized with AdamW [58] with a learning rate of 4e-6 and an epsilon of 1e-8. A linear scheduler is used and the warmup proportion is set to 1%.

For all datasets, the edge reasoning is performed in 2 hops. The edge-extraction threshold is set to 0.25. The edge repetition  $d$  is set to 2. The hidden sizes in GRU and perceptron are also set to

TABLE 4  
Experimental results on the MuTual development set. Recalls (R@1, R@2) and Mean Reciprocal Rank (MRR) are reported.

Method	MuTual			MuTual <sup>plus</sup>		
	R@1	R@2	MRR	R@1	R@2	MRR
Chance	25.00	50.00	60.40	25.00	50.00	60.40
TF-IDF	27.60	54.10	54.10	28.30	53.00	76.30
Dual LSTM [56]	26.60	52.80	53.80	-	-	-
SMN [8]	27.40	52.40	57.50	26.40	52.40	57.80
DAM [9]	23.90	46.30	57.50	26.10	52.00	64.50
<i>Transformer-based PLMs</i>						
BERT-Base [24]	65.70	86.70	80.30	51.40	78.70	71.50
RoBERTa-Base [25]	69.50	87.80	82.40	62.20	85.30	78.20
GPT-2 [49]	33.50	59.50	58.60	30.50	56.50	56.20
GPT-2-FT [49]	39.80	64.60	62.80	22.60	57.70	52.80
BERT-Base-MC [24]	66.10	87.10	80.60	58.60	79.10	75.10
RoBERTa-Base-MC [25]	69.30	88.70	82.50	62.10	83.00	77.80
RoBERTa-Large-MC [25]	85.10	94.47	91.63	73.25	91.76	85.11
<i>Dialogue Systems</i>						
Focal Reasoner (RoBERTa <sub>Base</sub> ) [51]	73.40	90.30	84.90	63.70	86.10	79.10
MDFN (RoBERTa <sub>Large</sub> ) [57]	84.50	95.30	91.40	-	-	-
MDFN (ELECTRA <sub>Large</sub> ) [57]	92.30	97.90	95.80	-	-	-
<i>Ours</i>						
DAGNs (RoBERTa <sub>Large</sub> )	86.79	96.50	92.73	78.22	92.55	88.14
DAGNs (ELECTRA <sub>Large</sub> )	92.55	98.19	95.97	82.73	95.26	90.51

1,024. The weight decay is set to 0.01 for all. The overall dropout rate is 10%. The model is trained for 30 epochs with a batch size of 16 on one Nvidia Tesla V100 GPU.

### 4.3 Results in Supervised Scenarios

#### 4.3.1 ReClor Dataset

**Compared Methods.** FastText [48] and Bi-LSTM learns semantics matching. FastText learns n-gram features for text classification, whereas Bi-LSTM learns contextual features with recurrent network architecture. Transformer-based pre-trained language models (PLMs) learn contextual embeddings from large-scale corpora. We also compare with the state-of-the-art Focal Reasoner [51], LReasoner [52], and MERIt [53]. The Focal Reasoner is a graph-based model that builds ad-hoc graphs with entity-based nodes and coreference edges. The LReasoner trains the PLMs with a contrastive learning framework, and the negative samples are constructed by pre-defined logical expressions. MERIt performs domain-specific pre-training also in a contrastive learning manner, where the augmented data is constructed via graph meta-paths. To conduct fair comparisons with LReasoner and MERIt, we train DAGNs by including the augmented negative data. The resulting models are denoted as “DAGNs + LReasoner” and “DAGNs + MERIt”, respectively. For “DAGNs + LReasoner”, logic graphs for the negative samples are constructed in the same manner. Then the model is fine-tuned with the contrastive learning objective function follows [52]. For “DAGNs + MERIt”, logic graphs for the negative instances are constructed as usual. The model is fine-tuned with the pre-trained checkpoints from [53]. The compared Focal Reasoner, LReasoner, MERIt, and the proposed DAGNs all use RoBERTa<sub>Large</sub> as the backbone PLM for a fair comparison.

**Results.** Table 2 demonstrates the results on the ReClor dataset. The DAGNs (RoBERTa<sub>Large</sub>) outperform Focal Reasoner (RoBERTa<sub>Large</sub>) in both ReClor and LogiQA, demonstrating the effectiveness of the logic graph-constrained learning. Moreover, the results of “DAGNs + LReasoner (RoBERTa<sub>Large</sub>)” and “DAGNs + MERIt (RoBERTa<sub>Large</sub>)” also outperform their counterparts. This

indicates that the structural constraints are still beneficial regardless of training schemes. Further, compared to the PLM counterpart RoBERTa<sub>Large</sub>, DAGNs (RoBERTa<sub>Large</sub>) show significant improvements. This indicates that the logic graphs provide useful information beyond the contextual embeddings learned from the plain texts, which is beneficial to reasoning. Moreover, the improvements on the test-HARD set are significant. DAGNs (RoBERTa<sub>Large</sub>) achieve 46.79%, which is comparable to the strong LReasoner (RoBERTa<sub>Large</sub>) and MERIt (RoBERTa<sub>Large</sub>) with augmented data. “DAGNs + LReasoner (RoBERTa<sub>Large</sub>)” and “DAGNs + MERIt (RoBERTa<sub>Large</sub>)” also show great improvements over LReasoner (RoBERTa<sub>Large</sub>) and MERIt (RoBERTa<sub>Large</sub>), respectively. The overall observations indicate the effectiveness of DAGNs and the structural logic representations are beneficial for challenging reasoning questions.

#### 4.3.2 LogiQA Dataset

**Compared Methods.** The word matching [54] and sliding window [55] perform lexical matching between the passage-question pair and candidate answers. Deep QA systems, including Stanford Attentive Reader [5], Gated-Attention Reader [6], and Co-Matching Network [7] calculate semantic similarity or use fine-grained attention mechanisms to match the context and the candidate answers. The performances are around chance, which indicates that the lexical or semantic matching is insufficient for catching the logic behind the texts. Transformer-based pre-trained language models (PLMs) perform better than lexical or semantic matching, but the results are still inferior. It is indicated that the powerful contextual embeddings partially help the logical reasoning QA, but the inferiority of the lack of logical structure is obvious. We also compare with the state-of-the-art methods Focal Reasoner [51], LReasoner [52], and MERIt [53]. Similar as in the ReClor dataset, the compared Focal Reasoner, LReasoner, MERIt, and the proposed DAGNs all use RoBERTa<sub>Large</sub> as the backbone PLM for a fair comparison.

TABLE 5

Zero-shot transfer between ReClor and LogiQA compared with supervised learning results. “RoBERTa-L” means “RoBERTa-Large”. “DAGNs-CT” indicates full training on the target dataset after zero-shot transfer.

Method	LogiQA → ReClor				ReClor → LogiQA	
	Dev	Test	Test-E	Test-H	Dev	Test
RoBERTa-L	41.40	38.30	41.82	35.54	35.79	37.94
DAGNs	44.20	41.90	46.59	38.21	41.47	39.94
DAGNs-CT	60.60	55.40	77.73	37.86	41.63	43.78

Method	ReClor				LogiQA	
	Dev	Test	Test-E	Test-H	Dev	Test
RoBERTa-L	62.60	55.60	75.50	40.00	35.02	35.33
DAGNs	66.80	61.00	79.09	46.79	39.63	42.09

TABLE 6

Zero-shot transfer from ReClor to MuTual and LogiQA to Mutual compared with supervised learning results. “DAGNs-CT” indicates full training on the target dataset after zero-shot transfer.

Method	ReClor → MuTual			ReClor → MuTual <sup>plus</sup>		
	R@1	R@2	MRR	R@1	R@2	MRR
RoBERTa-Large	41.31	69.64	64.61	37.25	63.43	61.21
DAGNs	48.53	74.60	70.56	45.37	71.90	67.67
DAGNs-CT	80.70	92.21	90.53	73.14	90.07	85.46

Method	LogiQA → MuTual			LogiQA → MuTual <sup>plus</sup>		
	R@1	R@2	MRR	R@1	R@2	MRR
RoBERTa-Large	25.96	51.58	58.00	21.44	48.53	55.68
DAGNs	58.24	81.38	76.94	48.42	75.73	70.14
DAGNs-CT	83.63	93.91	91.91	74.15	90.97	85.98

Method	MuTual			MuTual <sup>plus</sup>		
	R@1	R@2	MRR	R@1	R@2	MRR
RoBERTa-Large	85.10	94.47	91.63	73.25	91.76	85.11
DAGNs	86.79	96.50	92.73	78.22	92.55	88.14

**Results.** Table 3 shows the results on the LogiQA dataset. DAGNs (RoBERTa<sub>Large</sub>) also outperform the Focal Reasoner (RoBERTa<sub>Large</sub>) on the LogiQA test set. Moreover, The DAGNs (RoBERTa<sub>Large</sub>) also outperform the data-augmented LReasoner (RoBERTa<sub>Large</sub>) and MERIt (RoBERTa<sub>Large</sub>) on the test set. Furthermore, using the training paradigm in LReasoner, “DAGNs + LReasoner (RoBERTa<sub>Large</sub>)” also show superiority over LReasoner (RoBERTa<sub>Large</sub>) and MERIt (RoBERTa<sub>Large</sub>). The results demonstrate that this method is generally effective for logical reasoning questions, regardless of training paradigms. The logic graph constraint provides beneficial guidance to representation learning and is superior to augmented plain texts.

### 4.3.3 MuTual Dataset

**Compared Methods.** The TF-IDF, Dual LSTM [56], SMN [8], and DAM [9] conduct semantic text matching between dialogue context and candidate responses by using similarity of feature attention. According to the recall at positions 1 and 2, these methods select the correct responses by chance. The MRRs are also all lower than chance. This is not surprising considering the high lexical overlap between the context and the negative responses. For pre-trained LMs, GPT [23], and GPT-2 [49] perform as poorly as the text-matching methods, indicating that the generative models are inferior in reasoning. BERT [24] and RoBERTa [25] show

TABLE 7

Ablation of graph representation and structure on ReClor.

Method	ReClor			
	Dev	Test	Test-E	Test-H
DAGNs	66.80	61.00	79.09	46.79
<i>Graph representation</i>				
random node embeddings	60.60	56.40	76.36	40.71
<i>Graph structure</i>				
homogeneous variable edges	63.60	59.30	77.73	44.82
fully-connected edge linking	63.00	56.10	74.32	41.79
random edge linking	61.00	55.90	74.09	41.61
single edge types	62.80	57.70	75.00	44.11
clause nodes	63.40	56.60	75.23	41.96
sentence nodes	60.40	57.30	74.32	43.93

TABLE 8

Ablation of model components (ReClor dev and test accuracy (%)). ER indicates the edge-reasoning mechanism. GB indicates the global graph representation. NT indicates binary node types. VE indicates variable edges.

Method	ReClor			
	Dev	Test	Test-E	Test-H
DAGNs	66.80	61.00	79.09	46.79
w/o GB	62.00	60.10	77.73	46.25
w/o ER	66.80	59.80	78.64	45.00
w/o ER, GB	67.40	59.50	78.41	44.64
w/o NT	63.40	58.70	77.05	44.29
w/o NT, VE. [1]	65.20	58.20	76.14	44.11
w/o graph reasoning	55.20	52.00	74.77	34.11

better performances, especially the RoBERTa-Large model. We also compare with Focal Reasoner [51] and MDFN [57]. Focal Reasoner is a graph-based model with entity-based nodes and coreference relations. MDFN uses multiple attention masks to decouple the contextual representations in utterance-aware and speaker-aware manners, then fuse the representation with a gate. We follow MDFN to use RoBERTa<sub>Large</sub> and ELECTRA<sub>Large</sub> as the backbone PLMs for a fair comparison.

**Results.** Table 4 shows the compared results on the MuTual datasets. DAGNs surpass the compared methods, including graph-based model and attention mask-based decoupling-fusion network. The results demonstrate that our proposed method is effective for less formal text such as multi-turn dialogue.

## 4.4 Results in Zero-shot Scenarios

We conduct zero-shot transfer experiments among the three datasets to see whether the constructed logic graph structure helps the models with unseen logical reasoning questions. Considering the similarity between ReClor and LogiQA and the distinguishment of MuTual, we first train the models on LogiQA, then conduct direct testing on the ReClor development set and test set in a zero-shot manner, and vice versa. We then train the models on ReClor or LogiQA, respectively, then evaluate the MuTual development set in a zero-shot way. For further comparison, we conduct continue full training on the target datasets. The results are demonstrated in Table 5 and Table 6.

### 4.4.1 Zero-shot Transfer between ReClor and LogiQA

Comparing the results in the zero-shot setting and that in the supervised learning setting, it is surprising that the pre-trained LM



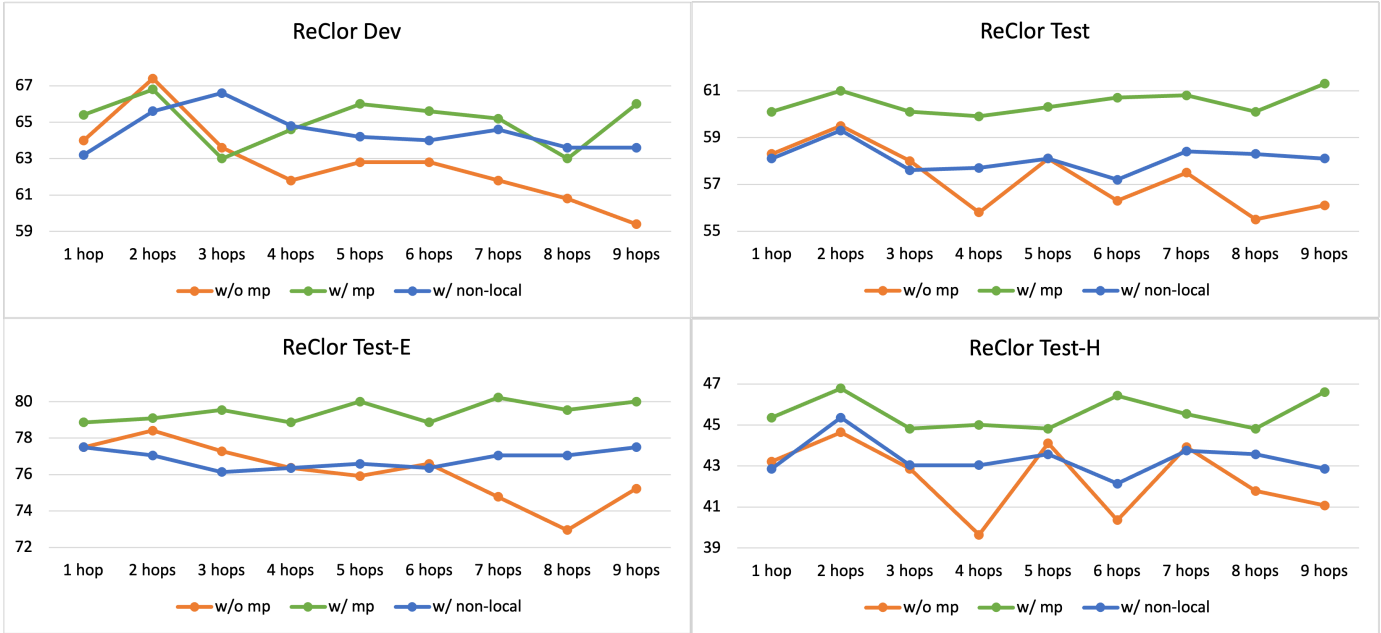


Fig. 5. Performance comparison among DAGNs, DAGNs with non-local GNNs, and DAGNs without the edge-reasoning mechanism over multiple GNN layers.

and our DAGNs both show generality to some extent. Transferring from LogiQA to ReClor, RoBERTa-Large reaches 38.30% in the test set, which is only 17.3 points behind that in the supervised learning setting. DAGNs (LogiQA → ReClor) achieve 41.90% in the test set compared to 59.50% in the supervised learning setting. Interestingly, the generality in the EASY subset is harder. Both PLM and DAGNs accuracies are around 40% in the zero-shot setting, but they achieve over 75% in the supervised learning setting. But the transfer in the HARD subset does not lose much. The performances in the zero-shot setting are over 35%, being comparable to the supervised-learning counterparts. Moreover, DAGNs (ReClor → LogiQA) achieve 41.47%/39.94% on the development and test sets, comparable to 39.63%/42.09% of the fine-tuning model. The experimental results indicate that the generality of DAGNs is better than RoBERTa-Large, both transferring from LogiQA to ReClor and from ReClor to LogiQA. It is indicated that the DAGNs improve the generality with the logic graphs and logic representations.

Moreover, after fine-tuning the zero-shot models on the target data, DAGNs-CT (ReClor → LogiQA) reaches 43.78% on the test set, DAGNs-CT (LogiQA → ReClor) reaches 55.40% on the test set, and over 30% on the test-EASY set, while the performance on test-HARD is only 0.35% inferior, which is still comparable. It is indicated that the transfer does not harm the performance given that the source and target data are different in reasoning types and data distribution.

#### 4.4.2 Zero-shot Transfer to MuTual

The RoBERTa-Large struggles with the transfer, especially from LogiQA. RoBERTa-Large (LogiQA → MuTual/MuTual<sup>plus</sup>) only achieves results around chance. This may be due to that the MuTual dataset shares less familiarity with the ReClor or LogiQA dataset, and the ReClor dataset is more challenging with more complex sentences and logical structures, so learning from ReClor makes solving the MuTual dataset easier. In contrast, the LogiQA provides less beneficial structural information for solving MuTual.

DAGNs (ReClor → MuTual/MuTual<sup>plus</sup>) outperform RoBERTa-Large (ReClor → MuTual/MuTual<sup>plus</sup>). Similar results are observed between DAGNs (LogiQA → MuTual/MuTual<sup>plus</sup>) and RoBERTa-Large (LogiQA → MuTual/MuTual<sup>plus</sup>). The improvements of transferring to MuTual<sup>plus</sup> are more significant than transferring to MuTual, which relieves the struggle of RoBERTa-Large. The observations are coherent with that in the ReClor → LogiQA setting. The results demonstrate that given MuTual/MuTual<sup>plus</sup> are significantly different from ReClor/LogiQA in data distribution and reasoning types, DAGNs show superiority in logical reasoning transfer.

Further fine-tuning on the MuTual/MuTual<sup>plus</sup> results in significant performance growth. R@1 of DAGNs-CT (ReClor/LogiQA → MuTual) are 80.70% and 83.63%, of DAGNs-CT (ReClor/LogiQA → MuTual<sup>plus</sup>) are 73.14% and 74.15%, respectively, which are comparable to their fine-tuning counterparts. The result improvements further demonstrate that DAGNs learn beneficial and general logic representations.

### 4.5 Ablation Study

We conduct an ablation study further to explore the benefits of each part of our model. We take a close look at the model components, the importance of the graph components, and the effect of GNN layer stacks.

#### 4.5.1 Importance of Graph Components

We further validate each graph component. Since the logic graph structure is significant to logical reasoning, we carefully modify the components of the logic graph and observe the performances. The results are shown in Table 7.

We first vary the graph representation. The node embeddings in DAGNs are initialized with the EDU embeddings merged from the contextual token embeddings. We modify the pre-trained and merged EDU features with randomly initialized embeddings. The development set accuracy drops to 60.60% and the test set to

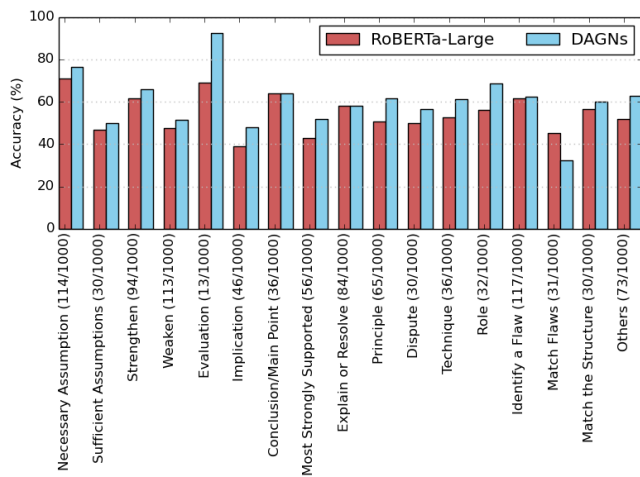


Fig. 6. Performance comparison on question types in the ReClor test set. The numbers in parenthesis mean the number of samples in each question type over the test set scale.

56.40%. It is worth noting that the accuracy in the HARD subset falls from 44.64% to 40.71%. It is a significant descent and demonstrates that the node features initialized from contextual embeddings are beneficial to logic graph reasoning.

We then vary the graph structure by modifying the edges and the nodes. We make two changes to the edges: (1) modifying the edge linking and (2) modifying the edge type. For edge linking, we first add variable edges within the context nodes and the candidate nodes (homogeneous variable edges), respectively. The performances drop to 63.60% on the development set, and 59.30% / 77.73% / 44.82% on the test / test-EASY / test-HARD sets, respectively. The results indicate that the homogeneous variable edges are redundant to the logic graphs. A possible reason is that the discourse connective edges within the context nodes and the candidate nodes are dense to some extent, so the homogeneous variable edges do not provide further information for the node feature update. Then, we ignore discourse relations and connect every pair of nodes, turning the graph fully connected. The resulting development accuracy drops to 63.00%, and test accuracy drops to 56.10%. Moreover, we remove all the edges from the logic graphs and randomly assign edges among the nodes with a Bernoulli distribution. The development set accuracy drops to 61.00%, and the test set the precision to 55.90%. The performances indicate that the fully-connected edge linking has unnecessary connections, while the random edge linking misses some linkings with helpful information. It reveals that in the logic graph we built, edges link EDUs in reasonable manners.

For uncovering the contribution of edge types, instead of the differentiation of explicit discourse relations and implicit ones, all edges are regarded as a single type. With a single edge type, the model reaches 62.80% on the development set and 57.70% on the test set, which is 4.6% and 1.8% inferior to the entire model. Therefore the two discourse-related edge types provide some helpful information to the model.

The nodes in the logic graphs act as reasoning units and are critical to logic representation learning. In substitution for EDUs, we use clauses or sentences as graph nodes. To obtain clause nodes, we remove “Explicit” connectives during discourse unit delimitation so that delimiters are only punctuation marks. For sentence nodes, we further reduce the delimiter library to solely

period (“.”). The development and test accuracies drop to 63.40% and 56.60% with the modified graphs with clause nodes. When replaced with coarser sentence nodes, the performance drops to 60.40% and 57.30%. This indicates that clause or sentence nodes carry less discourse information and act poorly as logical reasoning units.

#### 4.5.2 Model Components

To see the benefits of each component in the representation learning, we carefully remove them from the model, and the results on ReClor are shown in Table 8. We first remove the edge-reasoning mechanism, and the results drop to 66.80%/59.80%/78.64%/45.00% on the development/test/test-EASY/test-HARD sets, which indicates the effectiveness of the edge-reasoning mechanism. Then, we remove the global node representation, from both full DAGNs and the DAGNs without edge-reasoning mechanism. Further performance drops are observed. It is indicated that the global graph representation catches some logical consistency between the context and candidates. We then reduce the node types to only one type. As a result, the logic graphs only have a single node type. The dev accuracy drops dramatically from 67.40% to 63.40%. The test accuracy is slightly inferior, from 59.50% to 58.70%. Then, the variable edges are further removed from the model. The test accuracy further declines to 58.20%. Therefore, the logic graphs have reasonable structures for the logical reasoning QA task.

We further remove the whole graph reasoning operation. As a result, the hierarchical fusion is removed. The performance drops dramatically. It is indicated that the lack of graph reasoning leads to the absence of logic-aware features and degenerates the performance. It demonstrates the necessity of logical structures.

#### 4.5.3 Effect of GNN Layer Stacks

We change the number of the stacked GNN layers in our model to see how the graph reasoning steps affect the performances. We compare the performances between the full DAGNs and the model without the edge-reasoning mechanism. We run both the DAGNs with and without the edge-reasoning mechanism in this setting. Moreover, to compare the edge-reasoning mechanism with the non-local graph neural networks [59], [60] for solving the over-smoothing problem [61], [62] over the GNN layer stacks, we also compare with the DAGNs with non-local GNNs [59] as a replacement of the edge-reasoning mechanism. The results are demonstrated in Figure 5.

Overall, the results show that the full DAGNs with the edge-reasoning mechanism perform steadily over multiple GNN layers, while the model without the edge-reasoning mechanism shows fluctuation and deterioration when the GNN iteration grows. Specifically, the DAGNs without edge-reasoning mechanism reach peak performances with around two-step aggregation, after which decreases due to the general over-smoothing problem. In contrast, performance gains are observed in the full DAGNs, especially on the test and test-HARD sets. This indicates that shallow aggregation is insufficient for complex logical reasoning tasks, while the learnable edge-reasoning mechanism greatly relieves the over-smoothing problem in graph reasoning so that the model achieves deeper multi-hop reasoning as required. One of the reasons is that the soft edge propagation in the edge-reasoning mechanism reasons new edges, which provides shortcuts for meaningful multi-hop relations and then accelerates the effective node feature update. As a result, the graph model takes fewer iterations to learn the multi-hop relations.

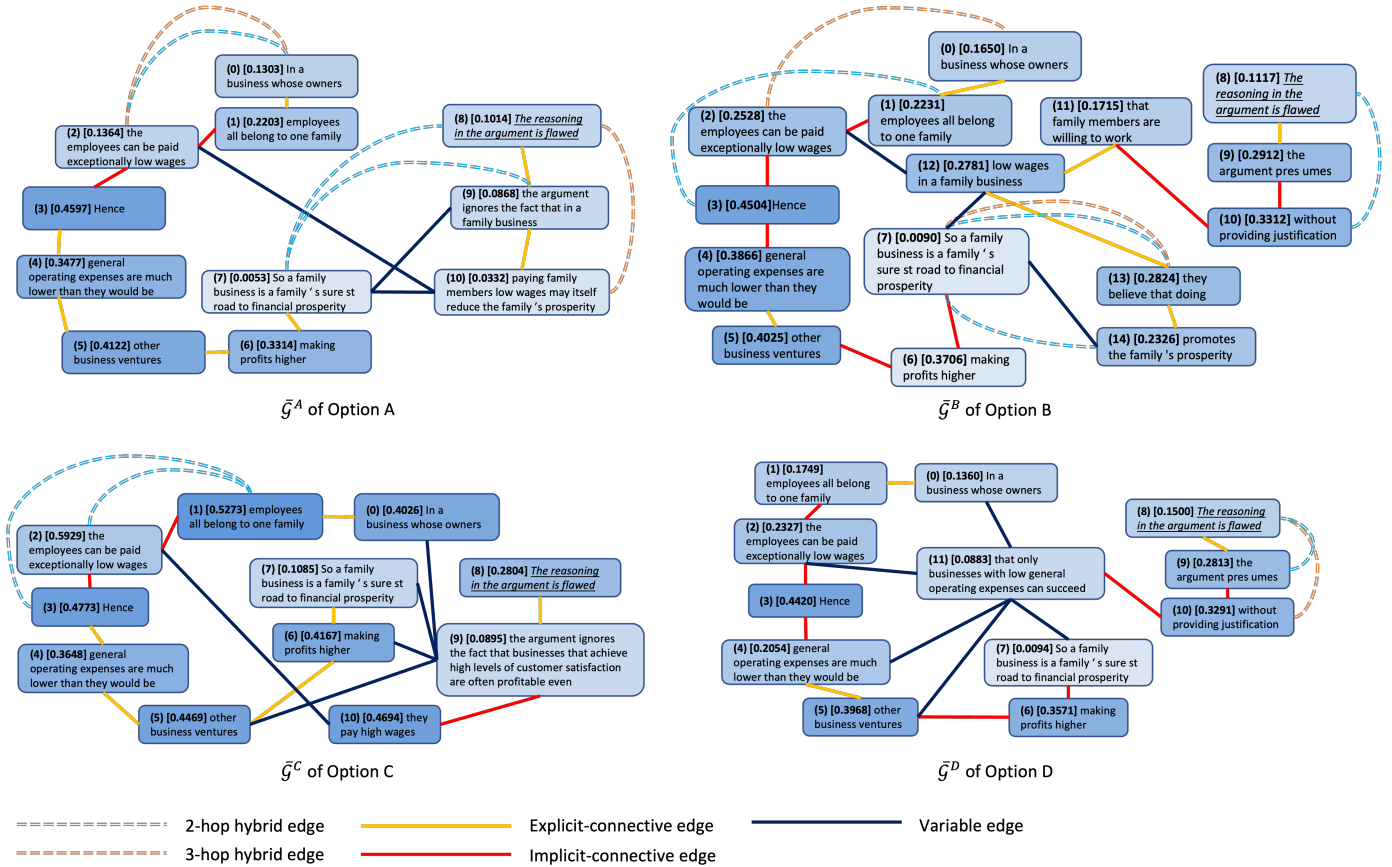


Fig. 7. Visualization of DAGNs with the learned hybrid edges and node weights. In this case, the correct answer is option A. The DAGNs give the correct answer.

Moreover, the DAGNs with non-local GNNs also show performance stability over multiple GNN layers, but are generally inferior to the DAGNs with the edge-reasoning mechanism. The results indicate that the edge-reasoning mechanism is superior in logical reasoning problems. The reason can be that the “attention-guided sorting” in the non-local GNNs pulls the distant nodes together according to a randomly initialized calibration vector, which is less informative than the edge propagation in the edge-reasoning mechanism to understand the logical relations.

### 4.6 Question Types

The ReClor dataset contains multiple question types corresponding to diverse logical reasoning capabilities. We evaluate models in each question type, and the results are demonstrated in Figure 6.

Generally speaking, DAGNs perform better on most types of problems. In question types such as “Evaluation”, “Technique”, and “Most Strongly Supported” that have high demands for knowledge of logical structures, the performance boosts over baseline models are significant. Therefore, the logic graphs are helpful to identify logical roles, such as the conclusion. Moreover, the questions of “Weakening” and “Implication” are extremely challenging, and DAGNs also achieve improvements. It is indicated that the constructed logic graphs provide weakening relation and entailment relation information. Other types of questions in which DAGNs perform well are “Strengthen”, “Conclusion/Main Point”, “Explain or Resolve”, “Principle” and so forth.

In three question types, “Match Flaws”, “Identify a Flaw”, and “Necessary Assumption”, DAGNs perform inferior to the RoBERTa-

Large, especially in the challenging “Match Flaws” questions. Therefore, although the logic graphs and representation learning are beneficial overall, they do not cover each logical reasoning type. The “Match Flaws” questions also require awareness of logical structures and paring the structures in the passage and the options. Since the texts are logically flawed, the logic graphs directly constructed from the texts are not logically sound. Hence the learned logic representations are less desirable.

### 4.7 Visualization

To further investigate the interpretability of the model, we visualize the generated hybrid edges and the learned node weights, respectively.

We first visualize the hybrid edges generated by the edge-reasoning mechanism, and two cases are shown in Figures 7 and 8. In option A in Figure 7, the edge-reasoning mechanism generates a 3-hop hybrid edge between node (8) and node (10), which bridges the question node and the key statement in the candidate. Moreover, the edge-reasoning mechanism learns the 2-hop edges (node (7), node (8)) and (node (7), node (9)). As node (7) is the conclusion, the edge-reasoning mechanism builds the hybrid edges for node (7) to help understand the argument and find the flaw. In contrast, in option C in this case, the key connections are lacking. Similarly, in the case of Figure 8, in option B, the model connects node (0) and node (4), which are the speaker and his/her opinion. The learned edges between node (9) and node (10), and between node (9) and node (11), builds connections between the speaker’s opinion, the



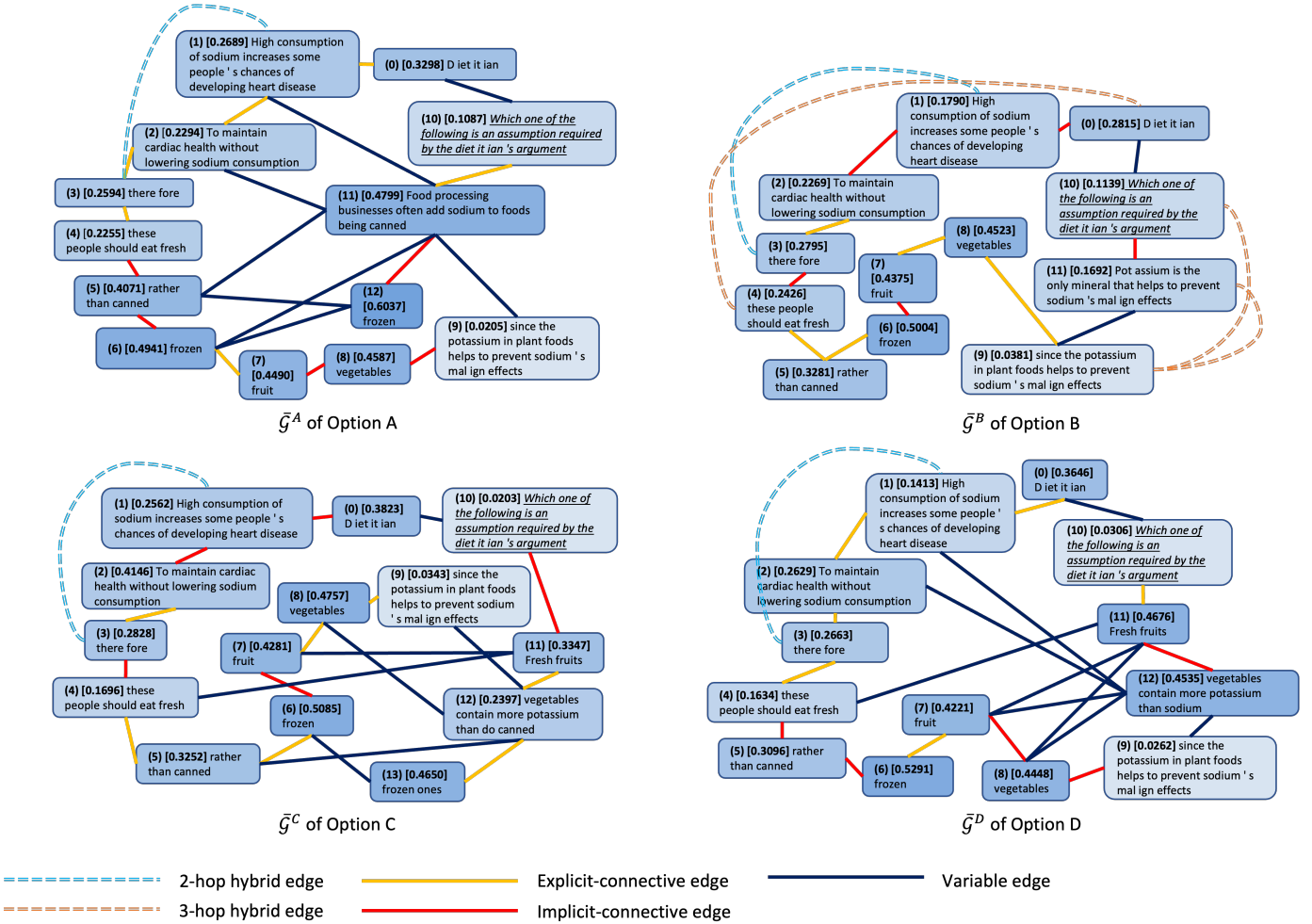


Fig. 8. Visualization of DAGNs with the learned hybrid edges and node weights. In this case, the correct answer is option C. The DAGNs give the correct answer.

question, and the assumption in the candidate answer, which help find the inconsistency between (9) and (11). In option C, the graph has dense connections, especially between the context nodes and candidate nodes. The generated hybrid edges are relatively few. A reason is that the constructed edges are sufficient for identifying the logical consistency.

Moreover, we visualize the graph node weights from multiple model variants, and two cases are presented in Figures 9 and 10 here. The node weights are the  $\alpha_i$  in Expression (7) demonstrated in Section 3.2.3 in the manuscript. The five model variants are (a) the full model DAGNs, (b) the DAGNs learned from LogiQA and then perform on the ReClor in a zero-shot manner, (c) the DAGNs without the edge-reasoning mechanism, (d) The DAGNs with fully-connected edge linking, and (e) the DAGNs with sentence nodes. In Figure 9, we observe that models (a) and (b) show better discrimination among the options as well as connections between the passage and the option. Interestingly, the zero-shot model (b) shows meaningful node attendance as the full-training model. Model (c) shows that without the learnable edge-reasoning mechanism, the model still being able to attend to the significant node such as node (3) that indicates an entailment, but the discrimination among the options is weaker. Model (d) shows that the model gives almost even attention to the sentences in the passage, nodes (2) and (5) have the highest weights, showing that the model is more interested in real entities and events, but it is less

aware of the conclusion or premise. Model (e) with sentence nodes still attends to the conclusion sentence, but the coarse-grained delimitation does not provide sufficient information for telling the correct answer. Similar observations are found in Figure 10. The model (d) with fully-connected edge linking fails the question with vague discrimination among the nodes.

## 5 RELATED WORKS

### 5.1 Textual Reasoning

Textual reasoning tasks such as reasoning QA [15], [63], [64], [65], Fact-Checking [66], and natural language inference (NLI) [67], [68] validate systems' reasoning with multiple schemes and granularity. Knowledge-based QA [63], [69], [70], [71], [72], [73], [74] provides large-scale knowledge bases [75], [76] for question answering. Multi-hop QA [14], [15] requires models to reason over multiple documents and find supporting facts for the question. Commonsense reasoning QA [64], [77], [78] requires reasoning out the unstated world knowledge behind it. Moreover, Fact-Checking [66] needs the models to retrieve supporting evidence for the given claims, while NLI [67], [68] requires the models to tell the inference relations between the given sentence pairs.

The previous QA [14], [15], [64] and Fact-Checking tasks [66] require models to retrieve supporting knowledge from a large set of documents. The models focus on effective knowledge retrieval and

<p><b>Option A</b> 🧠</p> <p>(0) [0.1303] In a business whose owners (1) [0.2203] employees all belong to one family (2) [0.1364] the employees can be paid exceptionally low wages (3) [0.4597] Hence (4) [0.3477] general operating expenses are much lower than they would be (5) [0.4122] other business ventures (6) [0.3314] making profits higher (7) [0.0053] So a family business is a family 's sure st road to financial prosperity (8) [0.1014] <i>The reasoning in the argument is flawed</i> (9) [0.0868] the argument ignores the fact that in a family business (10) [0.0332] paying family members low wages may itself reduce the family 's prosperity</p>	<p><b>Option B</b></p> <p>(0) [0.1650] In a business whose owners (1) [0.2231] employees all belong to one family (2) [0.2528] the employees can be paid exceptionally low wages (3) [0.4504] Hence (4) [0.3866] general operating expenses are much lower than they would be (5) [0.4025] other business ventures (6) [0.3706] making profits higher (7) [0.0090] So a family business is a family 's sure st road to financial prosperity (8) [0.1117] <i>The reasoning in the argument is flawed</i> (9) [0.2912] the argument pres umes (10) [0.3312] without providing justification (11) [0.1715] that family members are willing to work (12) [0.2781] low wages in a family business (13) [0.2824] they believe that doing (14) [0.2326] promotes the family 's prosperity</p>
<p><b>Option C</b></p> <p>(0) [0.4026] In a business whose owners (1) [0.5273] employees all belong to one family (2) [0.5929] the employees can be paid exceptionally low wages (3) [0.4773] Hence (4) [0.3648] general operating expenses are much lower than they would be (5) [0.4469] other business ventures (6) [0.4167] making profits higher (7) [0.1085] So a family business is a family 's sure st road to financial prosperity (8) [0.2804] <i>The reasoning in the argument is flawed</i> (9) [0.0895] the argument ignores the fact that businesses that achieve high levels of customer satisfaction are often profitable even (10) [0.4694] they pay high wages</p>	<p><b>Option D</b></p> <p>(0) [0.1360] In a business whose owners (1) [0.1749] employees all belong to one family (2) [0.2327] the employees can be paid exceptionally low wages (3) [0.4420] Hence (4) [0.2054] general operating expenses are much lower than they would be (5) [0.3968] other business ventures (6) [0.3571] making profits higher (7) [0.0094] So a family business is a family 's sure st road to financial prosperity (8) [0.1500] <i>The reasoning in the argument is flawed</i> (9) [0.2813] the argument pres umes (10) [0.3291] without providing justification (11) [0.0883] that only businesses with low general operating expenses can succeed</p>

(a) DAGNs

<p><b>Option A</b> 🧠</p> <p>(0) [0.1751] In a business whose owners (1) [0.2302] employees all belong to one family (2) [0.2151] the employees can be paid exceptionally low wages (3) [0.4541] Hence (4) [0.2835] general operating expenses are much lower than they would be (5) [0.3378] other business ventures (6) [0.4193] making profits higher (7) [0.0158] So a family business is a family 's sure st road to financial prosperity (8) [0.1111] <i>The reasoning in the argument is flawed</i> (9) [0.1544] the argument ignores the fact that in a family business (10) [0.0654] paying family members low wages may itself reduce the family 's prosperity</p>	<p><b>Option B</b> 🧠</p> <p>(0) [0.1630] In a business whose owners (1) [0.2289] employees all belong to one family (2) [0.2424] the employees can be paid exceptionally low wages (3) [0.4451] Hence (4) [0.2540] general operating expenses are much lower than they would be (5) [0.3329] other business ventures (6) [0.3398] making profits higher (7) [0.0079] So a family business is a family 's sure st road to financial prosperity (8) [0.0722] <i>The reasoning in the argument is flawed</i> (9) [0.2824] the argument pres umes (10) [0.4007] without providing justification (11) [0.3228] that family members are willing to work (12) [0.1999] low wages in a family business (13) [0.3977] they believe that doing (14) [0.1169] promotes the family 's prosperity</p>
<p><b>Option C</b></p> <p>(0) [0.1865] In a business whose owners (1) [0.2560] employees all belong to one family (2) [0.2938] the employees can be paid exceptionally low wages (3) [0.4482] Hence (4) [0.3351] general operating expenses are much lower than they would be (5) [0.3255] other business ventures (6) [0.4199] making profits higher (7) [0.0250] So a family business is a family 's sure st road to financial prosperity (8) [0.0715] <i>The reasoning in the argument is flawed</i> (9) [0.0048] the argument ignores the fact that businesses that achieve high levels of customer satisfaction are often profitable even (10) [0.2677] they pay high wages</p>	<p><b>Option D</b></p> <p>(0) [0.1524] In a business whose owners (1) [0.2341] employees all belong to one family (2) [0.2720] the employees can be paid exceptionally low wages (3) [0.4445] Hence (4) [0.2133] general operating expenses are much lower than they would be (5) [0.3242] other business ventures (6) [0.4170] making profits higher (7) [0.0196] So a family business is a family 's sure st road to financial prosperity (8) [0.0976] <i>The reasoning in the argument is flawed</i> (9) [0.3091] the argument pres umes (10) [0.4641] without providing justification (11) [0.0377] that only businesses with low general operating expenses can succeed</p>

(b) Zero-shot transfer (source: LogiQA)

<p><b>Option A</b> 🧠</p> <p>(0) [0.0322] In a business whose owners (1) [0.0057] employees all belong to one family (2) [0.0078] the employees can be paid exceptionally low wages (3) [0.3695] Hence (4) [0.0006] general operating expenses are much lower than they would be (5) [0.2096] other business ventures (6) [0.1452] making profits higher (7) [0.0000] So a family business is a family 's sure st road to financial prosperity (8) [0.1111] <i>The reasoning in the argument is flawed</i> (9) [0.0002] the argument ignores the fact that in a family business (10) [0.0004] paying family members low wages may itself reduce the family 's prosperity</p>	<p><b>Option B</b></p> <p>(0) [0.0316] In a business whose owners (1) [0.0052] employees all belong to one family (2) [0.0070] the employees can be paid exceptionally low wages (3) [0.3668] Hence (4) [0.0004] general operating expenses are much lower than they would be (5) [0.2288] other business ventures (6) [0.1461] making profits higher (7) [0.0000] So a family business is a family 's sure st road to financial prosperity (8) [0.0012] <i>The reasoning in the argument is flawed</i> (9) [0.0281] the argument pres umes (10) [0.0795] without providing justification (11) [0.0069] that family members are willing to work (12) [0.0136] low wages in a family business (13) [0.0725] they believe that doing (14) [0.0570] promotes the family 's prosperity</p>
<p><b>Option C</b></p> <p>(0) [0.0368] In a business whose owners (1) [0.0061] employees all belong to one family (2) [0.0075] the employees can be paid exceptionally low wages (3) [0.3619] Hence (4) [0.0006] general operating expenses are much lower than they would be (5) [0.2204] other business ventures (6) [0.1307] making profits higher (7) [0.0000] So a family business is a family 's sure st road to financial prosperity (8) [0.0015] <i>The reasoning in the argument is flawed</i> (9) [0.0000] the argument ignores the fact that businesses that achieve high levels of customer satisfaction are often profitable even (10) [0.0537] they pay high wages</p>	<p><b>Option D</b></p> <p>(0) [0.0361] In a business whose owners (1) [0.0055] employees all belong to one family (2) [0.0058] the employees can be paid exceptionally low wages (3) [0.3447] Hence (4) [0.0012] general operating expenses are much lower than they would be (5) [0.2142] other business ventures (6) [0.1726] making profits higher (7) [0.0001] So a family business is a family 's sure st road to financial prosperity (8) [0.0014] <i>The reasoning in the argument is flawed</i> (9) [0.0500] the argument pres umes (10) [0.1312] without providing justification (11) [0.0018] that only businesses with low general operating expenses can succeed</p>

(c) W/o edge-reasoning

<p><b>Option A</b></p> <p>(0) [0.3695] In a business whose owners (1) [0.2139] employees all belong to one family (2) [0.5311] the employees can be paid exceptionally low wages (3) [0.4867] Hence (4) [0.4703] general operating expenses are much lower than they would be (5) [0.5086] other business ventures (6) [0.3851] making profits higher (7) [0.0294] So a family business is a family 's sure st road to financial prosperity (8) [0.0280] <i>The reasoning in the argument is flawed</i> (9) [0.0282] the argument ignores the fact that in a family business (10) [0.0330] paying family members low wages may itself reduce the family 's prosperity</p>	<p><b>Option B</b> 🧠</p> <p>(0) [0.3991] In a business whose owners (1) [0.1888] employees all belong to one family (2) [0.4389] the employees can be paid exceptionally low wages (3) [0.4814] Hence (4) [0.3149] general operating expenses are much lower than they would be (5) [0.5090] other business ventures (6) [0.3668] making profits higher (7) [0.0109] So a family business is a family 's sure st road to financial prosperity (8) [0.0268] <i>The reasoning in the argument is flawed</i> (9) [0.1198] the argument pres umes (10) [0.1896] without providing justification (11) [0.2687] that family members are willing to work (12) [0.3086] low wages in a family business (13) [0.2751] they believe that doing (14) [0.4092] promotes the family 's prosperity</p>
<p><b>Option C</b></p> <p>(0) [0.4474] In a business whose owners (1) [0.2401] employees all belong to one family (2) [0.5922] the employees can be paid exceptionally low wages (3) [0.4907] Hence (4) [0.5383] general operating expenses are much lower than they would be (5) [0.5412] other business ventures (6) [0.4304] making profits higher (7) [0.0223] So a family business is a family 's sure st road to financial prosperity (8) [0.0182] <i>The reasoning in the argument is flawed</i> (9) [0.0001] the argument ignores the fact that businesses that achieve high levels of customer satisfaction are often profitable even (10) [0.0947] they pay high wages</p>	<p><b>Option D</b></p> <p>(0) [0.4848] In a business whose owners (1) [0.2392] employees all belong to one family (2) [0.3495] the employees can be paid exceptionally low wages (3) [0.4602] Hence (4) [0.5130] general operating expenses are much lower than they would be (5) [0.4654] other business ventures (6) [0.3852] making profits higher (7) [0.0697] So a family business is a family 's sure st road to financial prosperity (8) [0.0170] <i>The reasoning in the argument is flawed</i> (9) [0.0847] the argument pres umes (10) [0.1691] without providing justification (11) [0.0023] that only businesses with low general operating expenses can succeed</p>

(d) Fully-connected edge linking

<p><b>Option A</b></p> <p>(0) [0.0135] In a business whose owners employees all belong to one family the employees can be paid exceptionally low wages (1) [0.0470] Hence general operating expenses are much lower than they would be other business ventures making profits higher (2) [0.1982] So a family business is a family 's sure st road to financial prosperity (3) [0.0000] <i>The reasoning in the argument is flawed</i> the argument ignores the fact that in a family business paying family members low wages may itself reduce the family 's prosperity</p>	<p><b>Option B</b> 🧠</p> <p>(0) [0.0164] In a business whose owners employees all belong to one family the employees can be paid exceptionally low wages (1) [0.0284] Hence general operating expenses are much lower than they would be other business ventures making profits higher (2) [0.1269] So a family business is a family 's sure st road to financial prosperity (3) [0.0080] <i>The reasoning in the argument is flawed</i> the argument pres umes without providing justification that family members are willing to work low wages in a family business they believe that doing promotes the family 's prosperity</p>
<p><b>Option C</b></p> <p>(0) [0.0160] In a business whose owners employees all belong to one family the employees can be paid exceptionally low wages (1) [0.0369] Hence general operating expenses are much lower than they would be other business ventures making profits higher (2) [0.2538] So a family business is a family 's sure st road to financial prosperity (3) [0.0000] <i>The reasoning in the argument is flawed</i> the argument ignores the fact that businesses that achieve high levels of customer satisfaction are often profitable even they pay high wages</p>	<p><b>Option D</b></p> <p>(0) [0.0151] In a business whose owners employees all belong to one family the employees can be paid exceptionally low wages (1) [0.0246] Hence general operating expenses are much lower than they would be other business ventures making profits higher (2) [0.1474] So a family business is a family 's sure st road to financial prosperity (3) [0.0000] <i>The reasoning in the argument is flawed</i> the argument pres umes without providing justification that only businesses with low general operating expenses can succeed</p>

(e) Sentence node

Fig. 9. Visualization of node weights learned from five models: DAGNs, DAGNs without the edge-reasoning mechanism, DAGNs with fully-connected edge linking, DAGNs zero-shot transferred from LogiQA, and DAGNs with sentence nodes. In this case, the correct answer is option A. In the passage, the EDU indices (\*) in green are node delimitations from the full logic graph, and the indices in red are from the sentence nodes. The DAGNs, DAGNs w/o edge-reasoning give the correct answer.



<p><b>Option A</b></p> <p>(0) [0.3298] D diet it ian (1) [0.2689] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.2294] To maintain cardiac health without lowering sodium consumption (3) [0.2594] there fore (4) [0.2255] these people should eat fresh (5) [0.4071] rather than canned (6) [0.4941] frozen (7) [0.4490] fruit (8) [0.4587] vegetables (9) [0.0205] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.1087] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.4799] Food processing businesses often add sodium to foods being canned (12) [0.6037] frozen</p> <p><b>Option C</b></p> <p>(0) [0.3823] D diet it ian (1) [0.2562] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.4146] To maintain cardiac health without lowering sodium consumption (3) [0.2828] there fore (4) [0.1696] these people should eat fresh (5) [0.3252] rather than canned (6) [0.5085] frozen (7) [0.4281] fruit (8) [0.4757] vegetables (9) [0.0343] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0203] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.3347] Fresh fruits (12) [0.2397] vegetables contain more potassium than do canned (13) [0.4650] frozen ones</p>	<p><b>Option B</b></p> <p>(0) [0.2815] D diet it ian (1) [0.1790] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.2269] To maintain cardiac health without lowering sodium consumption (3) [0.2795] there fore (4) [0.2426] these people should eat fresh (5) [0.3281] rather than canned (6) [0.5004] frozen (7) [0.4375] fruit (8) [0.4523] vegetables (9) [0.0381] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.1139] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.1692] Pot assium is the only mineral that helps to prevent sodium ' s mal ign effects</p> <p><b>Option D</b></p> <p>(0) [0.3646] D diet it ian (1) [0.1413] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.2629] To maintain cardiac health without lowering sodium consumption (3) [0.2663] there fore (4) [0.1634] these people should eat fresh (5) [0.3096] rather than canned (6) [0.5291] frozen (7) [0.4221] fruit (8) [0.4448] vegetables (9) [0.0262] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0375] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.4676] Fresh fruits (12) [0.4535] vegetables contain more potassium than sodium</p>
--	--

(a) DAGNs

<p><b>Option A</b></p> <p>(0) [0.3613] D diet it ian (1) [0.0334] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.1273] To maintain cardiac health without lowering sodium consumption (3) [0.3414] there fore (4) [0.2030] these people should eat fresh (5) [0.2017] rather than canned (6) [0.4458] frozen (7) [0.4762] fruit (8) [0.3933] vegetables (9) [0.0044] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0026] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.0636] Food processing businesses often add sodium to foods being canned (12) [0.4426] frozen</p> <p><b>Option C</b></p> <p>(0) [0.3712] D diet it ian (1) [0.0473] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.2225] To maintain cardiac health without lowering sodium consumption (3) [0.3350] there fore (4) [0.2230] these people should eat fresh (5) [0.3482] rather than canned (6) [0.5892] frozen (7) [0.5165] fruit (8) [0.4846] vegetables (9) [0.0031] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0252] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.3909] Fresh fruits (12) [0.1304] vegetables contain more potassium than do canned (13) [0.5632] frozen ones</p>	<p><b>Option B</b></p> <p>(0) [0.3679] D diet it ian (1) [0.0376] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.1909] To maintain cardiac health without lowering sodium consumption (3) [0.3557] there fore (4) [0.2211] these people should eat fresh (5) [0.3312] rather than canned (6) [0.4873] frozen (7) [0.4740] fruit (8) [0.3904] vegetables (9) [0.0042] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0075] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.0164] Pot assium is the only mineral that helps to prevent sodium ' s mal ign effects</p> <p><b>Option D</b></p> <p>(0) [0.3746] D diet it ian (1) [0.0441] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.2113] To maintain cardiac health without lowering sodium consumption (3) [0.3442] there fore (4) [0.2081] these people should eat fresh (5) [0.2481] rather than canned (6) [0.4708] frozen (7) [0.4695] fruit (8) [0.4094] vegetables (9) [0.0037] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0376] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.3819] Fresh fruits (12) [0.2117] vegetables contain more potassium than sodium</p>
--	--

(b) Zero-shot transfer (source: LogiQA)

<p><b>Option A</b></p> <p>(0) [0.1071] D diet it ian (1) [0.0006] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.0083] To maintain cardiac health without lowering sodium consumption (3) [0.2275] there fore (4) [0.0557] these people should eat fresh (5) [0.1767] rather than canned (6) [0.3839] frozen (7) [0.4093] fruit (8) [0.3396] vegetables (9) [0.0018] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0009] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.0070] Food processing businesses often add sodium to foods being canned (12) [0.3406] frozen</p> <p><b>Option C</b></p> <p>(0) [0.1286] D diet it ian (1) [0.0010] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.0101] To maintain cardiac health without lowering sodium consumption (3) [0.2340] there fore (4) [0.1012] these people should eat fresh (5) [0.1532] rather than canned (6) [0.3629] frozen (7) [0.4209] fruit (8) [0.3657] vegetables (9) [0.0024] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0012] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.1571] Fresh fruits (12) [0.0045] vegetables contain more potassium than do canned (13) [0.1380] frozen ones</p>	<p><b>Option B</b></p> <p>(0) [0.1151] D diet it ian (1) [0.0019] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.0154] To maintain cardiac health without lowering sodium consumption (3) [0.2333] there fore (4) [0.0650] these people should eat fresh (5) [0.1399] rather than canned (6) [0.3482] frozen (7) [0.3963] fruit (8) [0.3317] vegetables (9) [0.0018] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0002] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.0000] Pot assium is the only mineral that helps to prevent sodium ' s mal ign effects</p> <p><b>Option D</b></p> <p>(0) [0.1164] D diet it ian (1) [0.0009] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.0103] To maintain cardiac health without lowering sodium consumption (3) [0.2257] there fore (4) [0.0765] these people should eat fresh (5) [0.1378] rather than canned (6) [0.3588] frozen (7) [0.4301] fruit (8) [0.3498] vegetables (9) [0.0013] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0008] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.1521] Fresh fruits (12) [0.0062] vegetables contain more potassium than sodium</p>
--	--

(c) W/o edge-reasoning

<p><b>Option A</b></p> <p>(0) [0.3260] D diet it ian (1) [0.0081] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.1307] To maintain cardiac health without lowering sodium consumption (3) [0.2942] there fore (4) [0.4515] these people should eat fresh (5) [0.4063] rather than canned (6) [0.4570] frozen (7) [0.4603] fruit (8) [0.4274] vegetables (9) [0.5027] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0080] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.1137] Food processing businesses often add sodium to foods being canned (12) [0.3874] frozen</p> <p><b>Option C</b></p> <p>(0) [0.3114] D diet it ian (1) [0.0103] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.1121] To maintain cardiac health without lowering sodium consumption (3) [0.2837] there fore (4) [0.4650] these people should eat fresh (5) [0.3703] rather than canned (6) [0.4208] frozen (7) [0.4727] fruit (8) [0.4253] vegetables (9) [0.6974] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0049] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.3523] Fresh fruits (12) [0.0739] vegetables contain more potassium than do canned (13) [0.2615] frozen ones</p>	<p><b>Option B</b></p> <p>(0) [0.3254] D diet it ian (1) [0.0130] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.1636] To maintain cardiac health without lowering sodium consumption (3) [0.2990] there fore (4) [0.4333] these people should eat fresh (5) [0.4019] rather than canned (6) [0.4756] frozen (7) [0.4632] fruit (8) [0.4390] vegetables (9) [0.7828] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0180] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.0082] Pot assium is the only mineral that helps to prevent sodium ' s mal ign effects</p> <p><b>Option D</b></p> <p>(0) [0.2995] D diet it ian (1) [0.0099] High consumption of sodium increases some people ' s chances of developing heart disease (2) [0.1367] To maintain cardiac health without lowering sodium consumption (3) [0.2726] there fore (4) [0.4744] these people should eat fresh (5) [0.4037] rather than canned (6) [0.4543] frozen (7) [0.4775] fruit (8) [0.4462] vegetables (9) [0.7433] since the potassium in plant foods helps to prevent sodium ' s mal ign effects (10) [0.0051] Which one of the following is an assumption required by the diet it ian ' s argument (11) [0.2988] Fresh fruits (12) [0.0587] vegetables contain more potassium than sodium</p>
--	--

(d) Fully-connected edge linking

<p><b>Option A</b></p> <p>(0) [0.3993] D diet it ian High consumption of sodium increases some people ' s chances of developing heart disease (1) [0.8498] To maintain cardiac health without lowering sodium consumption (2) [0.1100] there fore these people should eat fresh rather than canned frozen fruit vegetables since the potassium in plant foods helps to prevent sodium ' s mal ign effects (3) [0.0118] Which one of the following is an assumption required by the diet it ian ' s argument (4) [0.5017] Food processing businesses often add sodium to foods being canned frozen</p> <p><b>Option C</b></p> <p>(0) [0.2661] D diet it ian High consumption of sodium increases some people ' s chances of developing heart disease (1) [0.7427] To maintain cardiac health without lowering sodium consumption (2) [0.0993] there fore these people should eat fresh rather than canned frozen fruit vegetables since the potassium in plant foods helps to prevent sodium ' s mal ign effects (3) [0.2562] Which one of the following is an assumption required by the diet it ian ' s argument (4) [0.5044] Fresh fruits vegetables contain more potassium than do canned frozen ones</p>	<p><b>Option B</b></p> <p>(0) [0.3445] D diet it ian High consumption of sodium increases some people ' s chances of developing heart disease (1) [0.7890] To maintain cardiac health without lowering sodium consumption (2) [0.0430] there fore these people should eat fresh rather than canned frozen fruit vegetables since the potassium in plant foods helps to prevent sodium ' s mal ign effects (3) [0.0266] Which one of the following is an assumption required by the diet it ian ' s argument (4) [0.5028] Pot assium is the only mineral that helps to prevent sodium ' s mal ign effects</p> <p><b>Option D</b></p> <p>(0) [0.2841] D diet it ian High consumption of sodium increases some people ' s chances of developing heart disease (1) [0.8201] To maintain cardiac health without lowering sodium consumption (2) [0.1011] there fore these people should eat fresh rather than canned frozen fruit vegetables since the potassium in plant foods helps to prevent sodium ' s mal ign effects (3) [0.3457] Which one of the following is an assumption required by the diet it ian ' s argument (4) [0.5051] Fresh fruits vegetables contain more potassium than sodium</p>
--	--

(e) Sentence node

Fig. 10. Visualization of node weights learned from five models: DAGNs, DAGNs without the edge-reasoning mechanism, DAGNs with fully-connected edge linking, DAGNs zero-shot transferred from LogiQA, and DAGNs with sentence nodes. In this case, the correct answer is option a. In the passage, the EDU indices (\*) in green are node delimitations from the full logic graph, and the indices in red are from the sentence nodes. The DAGNs, DAGNs w/o edge-reasoning give the correct answer.



semantic matching. For example, HGN [12] constructs hierarchical graphs to aggregate clues from the different granularity of evidence such as paragraph selection and supporting fact extraction. GEAR [18] constructs fully-connected evidence graphs with evidence-claim pairs as nodes for claim verification. Such models do not uncover text structures or simulate the reasoning processes with a given document. In contrast, solving logical reasoning questions requires the models to first reconstruct the structural reasoning process behind the text, and identify the logical components and relations, after which they can answer the questions about conclusion, assumption, argumentation strength, and logical fallacy. To achieve this, DAGNs use discourse-aware graphs to identify the logical components and use the variable edges to simulate the patterns. As a result, the graph reasoning under the structural constraints focuses on logic feature updates. Moreover, the edge-reasoning mechanism adapts the logical relations during training for more general logic representations.

On the other hand, previous tasks focus on the awareness of commonsense and world knowledge. For example, KagNet [79] and MHGRN [80] encode subgraphs from ConceptNet [81] and learn entity-based relational paths to answer commonsense questions. K-Adapter [82] injects knowledge into pre-trained models. For solving NLI [67], DRCN [20] aggregates the semantics, while SemBERT [21] and SGNet [22] learn semantics under different linguistic constraints. In contrast, logical reasoning QA focus on inference patterns rather than knowledge. For example in Figure 1, the correct inference pattern is the law of contraposition: “if A implies B, then not-B implies not-A, and vice versa,” which is the key to the question. The law is true regardless of the details in A and B. Such knowledge-inference disentanglement provides generality to unseen reasoning data. To this end, DAGNs is a pilot study for modeling the inference structure rather than focusing on knowledge.

Moreover, the recent Focal Reasoner [51] for logical reasoning QA also performs graph reasoning. However, the constructed graphs extract entities and coreference relations following the previous QA models, which shows inferiority in capturing the logical relations between statements. Besides, LReasoner [52] trains the PLMs with a contrastive learning framework, and the negative samples are constructed by pre-defined logical expressions. The negative samples are derived by logical expressions. MERIt [53] performs domain-specific pre-training also in a contrastive learning manner, where the augmented data is constructed via graph meta-paths. However, the injected logic-biased data is in natural language format, and the model they use is plain PLM, which models the logical reasoning process implicitly. It remains unclear how explicit logic formulation facilitates QA systems and what kind of logical structure is beneficial. Hence, in contrast, this paper focuses on logic-biased deep models that explicitly model the logical reasoning process and obtain the desired logic features. Our method also leverages PLMs but does not use augmented data. Hence this study is orthogonal to the previous [52].

## 5.2 Discourse Applications

Discourse information provides a high-level understanding of texts and hence is beneficial for many natural language tasks, for instance, text summarization [83], [84], [85], [86], neural machine translation [87], and coherent text generation [88]. There are also discourse-based applications for reading comprehension. DISCERN [89] segments texts into EDUs and learns interactive EDU features.

Mihaylov and Frank [90] provide additional discourse-based annotations and encode them with discourse-aware self-attention models. However, such information is not yet considered in logical reasoning. Unlike previous works, this work builds discourse-aware logic graphs by first using discourse relations as graph edges that connect EDUs, then learning the discourse features via message passing with graph neural networks.

In natural language processing, the most influential theories of discourse structure are the Rhetorical Structure Theory (RST) [41] and Lexicalized Tree-Adjoining Grammar for Discourse (DLTAG) [91]. RST studies reconstructing tree-like structures for texts. The D-LTAG focuses on detecting discourse relations within local text units, and the units are disjoint sentences or two clauses in a sentence. Inspired by the theories, several treebanks are constructed, and the most influential ones are RST-DT [92] and PDTB [92]. Models [42], [93], [94], [95], [96] are trained on these treebanks to accomplish discourse parsing. And discourse parsing is also applied for downstream applications [89], [97], [98], [99].

However, current discourse parsers are primarily trained on small datasets via supervised learning, where the representative corpus is the 1 million-word Wall Street Journal (WSJ) Corpus. As a result, it is challenging for the parsers to transfer to unseen texts, especially in new topics or domains. Therefore, these parsers are not applicable for logical structure parsing. In this paper, we customize rules to perform discourse segmentation and relation detection based on observations of the argument passages.

## 6 CONCLUSION

This paper explores a structure-based solution to textual logical reasoning that explicitly models the logical reasoning process. The challenges include: (1) Uncovering the inference structure from plain texts for effective structural constraints. (2) Learning the inference processes rather than the knowledge for effective logical reasoning.

To address the problems, we propose discourse-aware graph networks (DAGNs) with logic graph construction and logic representation learning. To construct beneficial logical structures, DAGNs get inspired by logic theories and convert plain text into logic graphs via several factors. The in-line discourse connectives indicate logical relations; hence are applied as text delimiters and split passages into clause-like logical units. Then the recurring topic-related terms are detected. The graph edges are two folds: the discourse connectives indicate logical relations, and the variable connection simulates logical expression derivation.

For learning the logic features, DAGNs take the constructed graphs as input and perform soft edge selection and propagation to produce multi-hop hybrid relations. It then updates the node features via several steps of graph reasoning. The graph network leverages contextual encoding and learns the logic representations, which are then fused for downstream prediction.

Extensive experiments are conducted on two logical reasoning QA datasets and one multi-turn dialogue reasoning dataset. The results demonstrate the overall superiority of DAGNs. The constructed logic graph structure is reasonable, and the edge-reasoning mechanism helps learn general logic representations and improves model stability. The zero-shot transfer results show that DAGNs perform remarkably well on unseen reasoning questions, which indicates that the learned logic representations are general in reasoning and beyond knowledge.

## REFERENCES

- [1] Y. Huang, M. Fang, Y. Cao, L. Wang, and X. Liang, “DAGN: Discourse-aware graph network for logical reasoning,” in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, Jun. 2021, pp. 5848–5855. [Online]. Available: <https://www.aclweb.org/anthology/2021.naacl-main.467> 1, 10
- [2] W. Yu, Z. Jiang, Y. Dong, and J. Feng, “Reclor: A reading comprehension dataset requiring logical reasoning,” in *ICLR 2020 : Eighth International Conference on Learning Representations*, 2020. 1, 2, 8
- [3] J. Liu, L. Cui, H. Liu, D. Huang, Y. Wang, and Y. Zhang, “Logiqa: A challenge dataset for machine reading comprehension with logical reasoning,” *IJCAI 2020*, 2020. 1, 2, 8
- [4] L. Cui, Y. Wu, S. Liu, Y. Zhang, and M. Zhou, “Mutual: A dataset for multi-turn dialogue reasoning,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 1406–1416. 1, 2, 8
- [5] D. Chen, J. Bolton, and C. D. Manning, “A thorough examination of the cnn/daily mail reading comprehension task,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 2358–2367. 1, 8, 9
- [6] B. Dhingra, H. Liu, Z. Yang, W. Cohen, and R. Salakhutdinov, “Gated-attention readers for text comprehension,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 1832–1846. 1, 8, 9
- [7] S. Wang, M. Yu, J. Jiang, and S. Chang, “A co-matching model for multi-choice reading comprehension,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2018, pp. 746–751. 1, 9
- [8] Y. Wu, W. Wu, C. Xing, M. Zhou, and Z. Li, “Sequential matching network: A new architecture for multi-turn response selection in retrieval-based chatbots,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 496–505. [Online]. Available: <https://aclanthology.org/P17-1046> 1, 9, 10
- [9] X. Zhou, L. Li, D. Dong, Y. Liu, Y. Chen, W. X. Zhao, D. Yu, and H. Wu, “Multi-turn response selection for chatbots with deep attention matching network,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 1118–1127. [Online]. Available: <https://aclanthology.org/P18-1103> 1, 9, 10
- [10] N. De Cao, W. Aziz, and I. Titov, “Question answering by reasoning across documents with graph convolutional networks,” 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)*, 2019, pp. 2306–2317. 1
- [11] L. Qiu, Y. Xiao, Y. Qu, H. Zhou, L. Li, W. Zhang, and Y. Yu, “Dynamically fused graph network for multi-hop reasoning,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 6140–6150. 1
- [12] Y. Fang, S. Sun, Z. Gan, R. Pillai, S. Wang, and J. Liu, “Hierarchical graph network for multi-hop question answering,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 8823–8838. 1, 17
- [13] C. Zheng and P. Kordjamshidi, “Srlgrn: Semantic role labeling graph reasoning network,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 8881–8891. 1
- [14] J. Welbl, P. Stenetorp, and S. Riedel, “Constructing datasets for multi-hop reading comprehension across documents,” *Transactions of the Association for Computational Linguistics*, vol. 6, pp. 287–302, 2018. 1, 14
- [15] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. Cohen, R. Salakhutdinov, and C. D. Manning, “Hotpotqa: A dataset for diverse, explainable multi-hop question answering,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 2369–2380. 1, 14
- [16] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in *ICLR (Poster)*, 2016. 1
- [17] Q. Ran, Y. Lin, P. Li, J. Zhou, and Z. Liu, “Numnet: Machine reading comprehension with numerical reasoning,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 2474–2484. 1
- [18] J. Zhou, X. Han, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, “GEAR: Graph-based evidence aggregating and reasoning for fact verification,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 892–901. 1, 17
- [19] Z. Liu, C. Xiong, M. Sun, and Z. Liu, “Fine-grained fact verification with kernel graph attention network,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7342–7351. 1
- [20] S. Kim, I. Kang, and N. Kwak, “Semantic sentence matching with densely-connected recurrent and co-attentive information,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 6586–6593. 1, 17
- [21] Z. Zhang, Y. Wu, H. Zhao, Z. Li, S. Zhang, X. Zhou, and X. Zhou, “Semantics-aware bert for language understanding,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 9628–9635. 1, 2, 17
- [22] Z. Zhang, Y. Wu, J. Zhou, S. Duan, H. Zhao, and R. Wang, “Sg-net: syntax guided transformer for language representation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 1, 2, 17
- [23] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving language understanding by generative pre-training,” 2018. 2, 8, 10
- [24] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “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)*, 2019, pp. 4171–4186. 2, 8, 9, 10
- [25] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019. 2, 8, 9, 10
- [26] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov, and Q. V. Le, “Xlnet: Generalized autoregressive pretraining for language understanding,” in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 5754–5764. [Online]. Available: <https://proceedings.neurips.cc/paper/2019/hash/dc6a7e655d7e5840e66733e9ee67cc69-Abstract.html> 2
- [27] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, “ALBERT: A lite BERT for self-supervised learning of language representations,” in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. [Online]. Available: <https://openreview.net/forum?id=H1eA7AEtV5> 2
- [28] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, “Glue: A multi-task benchmark and analysis platform for natural language understanding,” *arXiv preprint arXiv:1804.07461*, 2018. 2
- [29] G. Lai, Q. Xie, H. Liu, Y. Yang, and E. Hovy, “Race: Large-scale reading comprehension dataset from examinations,” in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 785–794. 2
- [30] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, “Squad: 100,000+ questions for machine comprehension of text,” in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 2383–2392. 2
- [31] E. Reif, A. Yuan, M. Wattenberg, F. B. Viegas, A. Coenen, A. Pearce, and B. Kim, “Visualizing and measuring the geometry of bert,” *Advances in Neural Information Processing Systems*, vol. 32, 2019. 2
- [32] D. Ye, Y. Lin, J. Du, Z. Liu, P. Li, M. Sun, and Z. Liu, “Coreferential Reasoning Learning for Language Representation,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 7170–7186. [Online]. Available: <https://aclanthology.org/2020.emnlp-main.582> 2
- [33] P. Clark, O. Tafjord, and K. Richardson, “Transformers as soft reasoners over language,” *arXiv preprint arXiv:2002.05867*, 2020. 2
- [34] R. Prasad, N. Dinesh, A. Lee, E. Mitsakaki, L. Robaldo, A. K. Joshi, and B. L. Webber, “The penn discourse treebank 2.0,” in *LREC*. Citeseer, 2008. 2, 3, 4
- [35] S. E. Toulmin, *The uses of argument*. Cambridge university press, 2003. 2, 3, 4
- [36] J. B. Freeman, “Argument structure: Representation and theory,” in *Argumentation Library*, 2011. 2, 3, 4
- [37] S. Yun, M. Jeong, R. Kim, J. Kang, and H. J. Kim, “Graph transformer



- networks,” *Advances in Neural Information Processing Systems*, vol. 32, pp. 11 983–11 993, 2019. **2**
- [38] M. Walicki, *Introduction To Mathematical Logic (Extended Edition)*. World Scientific Publishing Company, 2016. **3**
- [39] C. Dutilh Novaes, “Argument and Argumentation,” in *The Stanford Encyclopedia of Philosophy*, Fall 2021 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2021. **3**
- [40] L. Groarke, “Informal Logic,” in *The Stanford Encyclopedia of Philosophy*, Fall 2021 ed., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2021. **3**
- [41] W. C. Mann and S. A. Thompson, “Rhetorical structure theory: Toward a functional theory of text organization,” *Text*, vol. 8, no. 3, pp. 243–281, 1988. **3, 17**
- [42] V. W. Feng and G. Hirst, “Two-pass discourse segmentation with pairing and global features,” *arXiv preprint arXiv:1407.8215*, 2014. **4, 17**
- [43] J. Li, A. Sun, and S. R. Joty, “Segbot: A generic neural text segmentation model with pointer network,” in *IJCAI*, 2018, pp. 4166–4172. **4**
- [44] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017. **7**
- [45] J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” *stat*, vol. 1050, p. 21, 2016. **7**
- [46] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, “On the properties of neural machine translation: Encoder-decoder approaches,” *arXiv preprint arXiv:1409.1259*, 2014. **7**
- [47] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. **7**
- [48] A. Joulin, É. Grave, P. Bojanowski, and T. Mikolov, “Bag of tricks for efficient text classification,” in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 2017, pp. 427–431. **8, 9**
- [49] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, “Language models are unsupervised multitask learners,” *OpenAI blog*, vol. 1, no. 8, p. 9, 2019. **8, 9, 10**
- [50] Y. Cui, W. Che, T. Liu, B. Qin, Z. Yang, S. Wang, and G. Hu, “Pre-training with whole word masking for chinese bert,” *arXiv preprint arXiv:1906.08101*, 2019. **8**
- [51] S. Ouyang, Z. Zhang, and H. Zhao, “Fact-driven logical reasoning,” *arXiv preprint arXiv:2105.10334*, 2021. **8, 9, 10, 17**
- [52] S. Wang, W. Zhong, D. Tang, Z. Wei, Z. Fan, D. Jiang, M. Zhou, and N. Duan, “Logic-driven context extension and data augmentation for logical reasoning of text,” *arXiv preprint arXiv:2105.03659*, 2021. **8, 9, 17**
- [53] F. Jiao, Y. Guo, X. Song, and L. Nie, “MERIt: Meta-Path Guided Contrastive Learning for Logical Reasoning,” in *Findings of the Association for Computational Linguistics: ACL 2022*. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 3496–3509. [Online]. Available: <https://aclanthology.org/2022.findings-acl.276> **8, 9, 17**
- [54] S. W.-t. Yih, M.-W. Chang, C. Meek, and A. Pastusiak, “Question answering using enhanced lexical semantic models,” 2013. **8, 9**
- [55] M. Richardson, C. J. Burges, and E. Renshaw, “MCTest: A challenge dataset for the open-domain machine comprehension of text,” in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, Washington, USA: Association for Computational Linguistics, Oct. 2013, pp. 193–203. [Online]. Available: <https://aclanthology.org/D13-1020> **8, 9**
- [56] R. Lowe, N. Pow, I. Serban, and J. Pineau, “The Ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems,” in *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. Prague, Czech Republic: Association for Computational Linguistics, Sep. 2015, pp. 285–294. [Online]. Available: <https://aclanthology.org/W15-4640> **9, 10**
- [57] L. Liu, Z. Zhang, H. Zhao, X. Zhou, and X. Zhou, “Filling the gap of utterance-aware and speaker-aware representation for multi-turn dialogue,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 15, pp. 13 406–13 414, May 2021. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/17582> **8, 9, 10**
- [58] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. [Online]. Available: <https://openreview.net/forum?id=Bkg6RiCqY7> **8**
- [59] M. Liu, Z. Wang, and S. Ji, “Non-local graph neural networks,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 12, pp. 10 270–10 276, 2021. **12**
- [60] C. Yu, Y. Liu, C. Gao, C. Shen, and N. Sang, “Representative graph neural network,” in *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*. Springer, 2020, pp. 379–396. **12**
- [61] Q. Li, Z. Han, and X.-M. Wu, “Deeper insights into graph convolutional networks for semi-supervised learning,” in *Thirty-Second AAAI conference on artificial intelligence*, 2018. **12**
- [62] W. Huang, Y. Rong, T. Xu, F. Sun, and J. Huang, “Tackling over-smoothing for general graph convolutional networks,” *arXiv e-prints*, pp. arXiv–2008, 2020. **12**
- [63] J. Berant, A. Chou, R. Frostig, and P. Liang, “Semantic parsing on freebase from question-answer pairs,” in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1533–1544. **14**
- [64] A. Talmor, J. Herzig, N. Lourie, and J. Berant, “CommonsenseQA: A question answering challenge targeting commonsense knowledge,” in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4149–4158. [Online]. Available: <https://aclanthology.org/N19-1421> **14**
- [65] D. Dua, Y. Wang, P. Dasigi, G. Stanovsky, S. Singh, and M. Gardner, “Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs,” 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)*, 2019, pp. 2368–2378. **14**
- [66] J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal, “Fever: a large-scale dataset for fact extraction and verification,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 809–819. **14**
- [67] S. Bowman, G. Angeli, C. Potts, and C. D. Manning, “A large annotated corpus for learning natural language inference,” in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 632–642. **14, 17**
- [68] A. Williams, N. Nangia, and S. R. Bowman, “A broad-coverage challenge corpus for sentence understanding through inference,” in *NAACL-HLT*, 2018. **14**
- [69] J. Bao, N. Duan, Z. Yan, M. Zhou, and T. Zhao, “Constraint-based question answering with knowledge graph,” in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. Osaka, Japan: The COLING 2016 Organizing Committee, Dec. 2016, pp. 2503–2514. [Online]. Available: <https://aclanthology.org/C16-1236> **14**
- [70] W.-t. Yih, M. Richardson, C. Meek, M.-W. Chang, and J. Suh, “The value of semantic parse labeling for knowledge base question answering,” in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 201–206. [Online]. Available: <https://aclanthology.org/P16-2033> **14**
- [71] A. Talmor and J. Berant, “The web as a knowledge-base for answering complex questions,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 641–651. [Online]. Available: <https://aclanthology.org/N18-1059> **14**
- [72] V. Lopez, C. Unger, P. Cimiano, and E. Motta, “Evaluating question answering over linked data,” *Journal of Web Semantics*, vol. 21, pp. 3–13, 2013. **14**
- [73] P. Trivedi, G. Maheshwari, M. Dubey, and J. Lehmann, “Lc-quad: A corpus for complex question answering over knowledge graphs,” in *International Semantic Web Conference*. Springer, 2017, pp. 210–218. **14**
- [74] M. Dubey, D. Banerjee, A. Abdelkawi, and J. Lehmann, “Lc-quad 2.0: A large dataset for complex question answering over wikidata and dbpedia,” in *International semantic web conference*. Springer, 2019, pp. 69–78. **14**
- [75] Google, “Freebase data dumps,” <https://developers.google.com/freebase/data,year%20>. **14**
- [76] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer *et al.*, “Dbpedia—a large-scale, multilingual knowledge base extracted from wikipedia,” *Semantic web*, vol. 6, no. 2, pp. 167–195, 2015. **14**
- [77] L. Huang, R. Le Bras, C. Bhagavatula, and Y. Choi, “Cosmos qa: Machine reading comprehension with contextual commonsense reasoning,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on*

- Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 2391–2401. 14
- [78] N. Tandon, B. Dalvi, K. Sakaguchi, P. Clark, and A. Bosselut, “Wiq: A dataset for “what if...” reasoning over procedural text,” 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)*, 2019, pp. 6078–6087. 14
- [79] B. Y. Lin, X. Chen, J. Chen, and X. Ren, “KagNet: Knowledge-aware graph networks for commonsense reasoning,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 2829–2839. [Online]. Available: <https://aclanthology.org/D19-1282> 17
- [80] Y. Feng, X. Chen, B. Y. Lin, P. Wang, J. Yan, and X. Ren, “Scalable multi-hop relational reasoning for knowledge-aware question answering,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 1295–1309. [Online]. Available: <https://aclanthology.org/2020.emnlp-main.99> 17
- [81] R. Speer, J. Chin, and C. Havasi, “Conceptnet 5.5: An open multilingual graph of general knowledge,” in *Thirty-first AAAI conference on artificial intelligence*, 2017. 17
- [82] R. Wang, D. Tang, N. Duan, Z. Wei, X. Huang, J. Ji, G. Cao, D. Jiang, and M. Zhou, “K-adapter: Infusing knowledge into pre-trained models with adapters,” in *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, ser. Findings of ACL, C. Zong, F. Xia, W. Li, and R. Navigli, Eds., vol. ACL/IJCNLP 2021. Association for Computational Linguistics, 2021, pp. 1405–1418. [Online]. Available: <https://doi.org/10.18653/v1/2021.findings-acl.121> 17
- [83] A. Cohan, F. Deroncourt, D. S. Kim, T. Bui, S. Kim, W. Chang, and N. Goharian, “A discourse-aware attention model for abstractive summarization of long documents,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 615–621. 17
- [84] S. Joty, G. Carenini, R. Ng, and G. Murray, “Discourse analysis and its applications,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 12–17. 17
- [85] J. Xu, Z. Gan, Y. Cheng, and J. Liu, “Discourse-aware neural extractive text summarization,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 5021–5031. 17
- [86] X. Feng, X. Feng, B. Qin, X. Geng, and T. Liu, “Dialogue discourse-aware graph convolutional networks for abstractive meeting summarization,” *arXiv preprint arXiv:2012.03502*, 2020. 17
- [87] E. Voita, P. Serdyukov, R. Sennrich, and I. Titov, “Context-aware neural machine translation learns anaphora resolution,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 1264–1274. 17
- [88] A. Bosselut, A. Celikyilmaz, X. He, J. Gao, P.-S. Huang, and Y. Choi, “Discourse-aware neural rewards for coherent text generation,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 173–184. 17
- [89] Y. Gao, C.-S. Wu, J. Li, S. Joty, S. C. Hoi, C. Xiong, I. King, and M. Lyu, “Discern: Discourse-aware entailment reasoning network for conversational machine reading,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 2439–2449. 17
- [90] T. Mihaylov and A. Frank, “Discourse-aware semantic self-attention for narrative reading comprehension,” 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)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 2541–2552. 17
- [91] B. Webber, “D-Itag: extending lexicalized tag to discourse,” *Cognitive Science*, vol. 28, no. 5, pp. 751–779, 2004. 17
- [92] L. Carlson, D. Marcu, and M. E. Okurowski, “Building a discourse-tagged corpus in the framework of rhetorical structure theory,” in *Current and new directions in discourse and dialogue*. Springer, 2003, pp. 85–112. 17
- [93] K. Hayashi, T. Hirao, and M. Nagata, “Empirical comparison of dependency conversions for rst discourse trees,” in *Proceedings of the 17th annual meeting of the special interest group on discourse and dialogue*, 2016, pp. 128–136. 17
- [94] Y. Ji and J. Eisenstein, “Representation learning for text-level discourse parsing,” in *Proceedings of the 52nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, 2014, pp. 13–24. 17
- [95] Y. Kishimoto, Y. Murawaki, and S. Kurohashi, “Adapting bert to implicit discourse relation classification with a focus on discourse connectives,” in *Proceedings of The 12th Language Resources and Evaluation Conference*, 2020, pp. 1152–1158. 17
- [96] W. Lei, Y. Xiang, Y. Wang, Q. Zhong, M. Liu, and M.-Y. Kan, “Linguistic properties matter for implicit discourse relation recognition: Combining semantic interaction, topic continuity and attribution,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018. 17
- [97] Y. Ji and N. A. Smith, “Neural discourse structure for text categorization,” in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2017, pp. 996–1005. 17
- [98] Y. Liu and M. Lapata, “Learning structured text representations,” *Transactions of the Association for Computational Linguistics*, vol. 6, pp. 63–75, 2018. 17
- [99] Y. Liu, I. Titov, and M. Lapata, “Single document summarization as tree induction,” 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)*, 2019, pp. 1745–1755. 17





**Yinya Huang** is currently a Ph.D. student in Computer Science at the School of Intelligent Systems Engineering at Sun Yat-Sen University, China. Her research interests include machine reasoning and natural language understanding.



**Liang Lin** is a full professor of Computer Science at Sun Yat-sen University and CEO of Darker-Matter AI. He worked as the Executive Director of the SenseTime Group from 2016 to 2018, leading the R&D teams in developing cutting-edge, deliverable solutions in computer vision, data analysis and mining, and intelligent robotic systems. He has authored or co-authored more than 200 papers in leading academic journals and conferences. He is an associate editor of IEEE Trans. Human-Machine Systems and IET

Computer Vision, and he served as the area/session chair for numerous conferences such as CVPR, ICME, ICCV. He was the recipient of Annual Best Paper Award by Pattern Recognition (Elsevier) in 2018, Dimond Award for best paper in IEEE ICME in 2017, ACM NPAR Best Paper Runners-Up Award in 2010, Google Faculty Award in 2012, award for the best student paper in IEEE ICME in 2014, and Hong Kong Scholars Award in 2014. He is a Fellow of IET.



**Lemao Liu** is a senior researcher at Natural Language Processing Center, Tencent AI Lab, China. Previously, he was with the National Institute of Information and Communications Technology (NICT), Japan. He received his Ph.D. degree from the Harbin Institute of Technology. His research interests include machine translation, syntactic parsing, and natural language understanding. He has published about 50 research papers in leading conferences and journals, such as ACL, EMNLP, NAACL, COLING, ICLR, AACL, and JAIR.

He received an outstanding paper award in ACL 2021 and the best demo award in CCL 2020. He served as a publication co-chair in EMNLP 2020 (Findings), and a senior program committee member in IJCAI 2021.



**Kun Xu** is currently a principal research scientist at Huawei. He was a senior research scientist in Tencent AI Lab from 2018 to 2021. He received his Ph.D. degree from Peking University in 2016. He has published more than 40 papers in top conferences of NLP such as ACL, EMNLP, and NAACL. He was the recipient of the best paper award in NAACL in 2021. His research interests include question-answering and semantic parsing.



**Xiaodan Liang** is currently an Associate Professor at Sun Yat-sen University. She was a postdoc researcher in the machine learning department at Carnegie Mellon University, working with Prof. Eric Xing, from 2016 to 2018. She received her Ph.D. degree from Sun Yat-sen University in 2016. She has published several cutting-edge projects on human-related analysis, including human parsing, pedestrian detection, and instance segmentation, 2D/3D human pose estimation, and activity recognition.



**Meng Fang** is currently an Assistant Professor at the University of Liverpool. He received his Ph.D. degree from the University of Technology Sydney, Australia, in 2015. He was a postdoctoral research fellow in the School of Computing and Information Systems, the University of Melbourne. His current research interests include reinforcement learning, natural language processing, and machine learning.