

Scene Restoring for Narrative Machine Reading Comprehension

Zhixing Tian^{1,2}, Yuanzhe Zhang¹, Kang Liu^{1,2}, Jun Zhao^{1,2},
Yantao Jia³, Zhicheng Sheng³

¹ National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing, 100190, China

² University of Chinese Academy of Sciences, Beijing, 100049, China

³ Huawei Technologies Co., Ltd, Beijing, 100085, China

{zhixing.tian, yzzhang, kliu, jzhao}@nlpr.ia.ac.cn

jamaths.h@163.com, shengzhicheng@huawei.com

Abstract

This paper focuses on machine reading comprehension for narrative passages. Narrative passages usually describe a chain of events. When reading this kind of passage, humans tend to restore a scene according to the text with their prior knowledge, which helps them understand the passage comprehensively. Inspired by this behavior of humans, we propose a method to let the machine imagine a scene during reading narrative for better comprehension. Specifically, we build a scene graph by utilizing Atomic as the external knowledge and propose a novel Graph Dimensional-Iteration Network (GDIN) to encode the graph. We conduct experiments on the ROCStories, a dataset of Story Cloze Test (SCT), and CosmosQA, a dataset of multiple choice. Our method achieves state-of-the-art.

1 Introduction

Machine Reading Comprehension (MRC) is an NLP task designed to evaluate a machine's ability to understand human language. This direction has recently drawn much attention due to the fast development of deep learning techniques and large-scale datasets. As a basic form of MRC, the comprehension of narrative has attracted long-standing interests (Mostafazadeh et al., 2017; Kočiský et al., 2018; Cui et al., 2019). In this paper, we focus on this kind of MRC task.

Unlike the other type of text, the narratives usually present a series of events, which are related to a scene in real life. In the field of perception, the scene is a kind of information that flows from a physical environment into a perceptual system (Ruderman and Bialek, 1993). When reading a narrative instead of being in a physical environment, humans tend to restore the scene in their mind according to the text with their prior knowledge for better perception and comprehension (Bower and

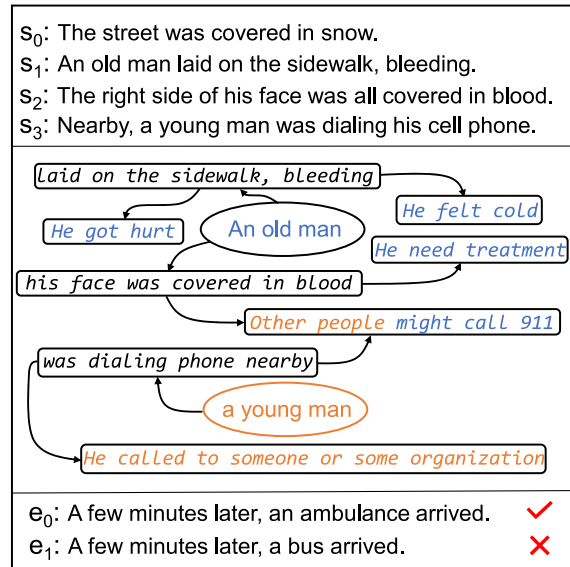


Figure 1: An example of Narrative MRC. Specifically, it is an example of Story Cloze Test (SCT), where given the first four sentences (s_0, s_1, s_2, s_3) of the story, a model is required to select the suitable ending from the candidates (e_0, e_1). The middle part is a presumable description of the scene restored by a human reader.

Morrow, 1990; Zwaan et al., 1995). The scene restored is an immediate association about the event, and it could be composed of the event itself, the state of the person roles, the possible cause and effect, and so on. To approach human intelligence, an MRC model is supposed to have a similar ability to restore the scene. However, previous work (Wang et al., 2016; Cui et al., 2019; Zhou et al., 2019a) pay little attention to this ability of narrative MRC models.

Figure 1 is an example of Narrative MRC. While reading the story sentence by sentence, a human tends to restore a scene in his or her mind as described in the figure. Subsequently, the human reader can infer that the suitable ending is e_0 based on information of the scene. Unfortunately, a ma-

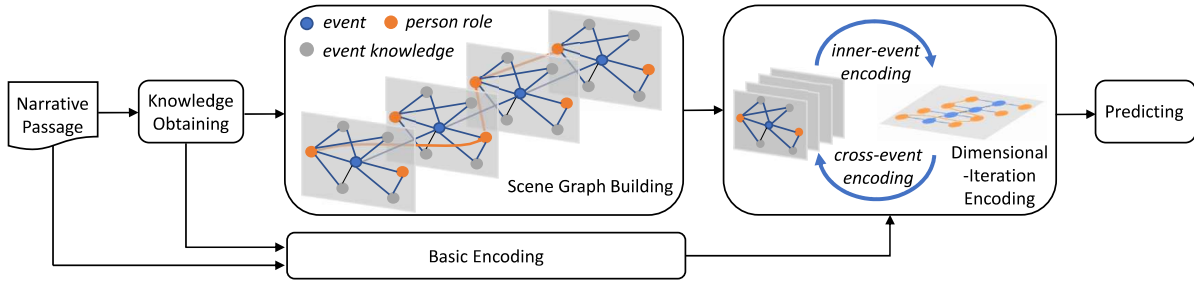


Figure 2: The basic overview of the proposed method

chine reader is not endowed with the ability to associate the prior knowledge, and cannot restore the scene according to the original narrative. As a result, It cannot thoroughly understand what happens in the story and possibly make a wrong decision. To address this problem, we proposed a novel method, which can restore the scene and utilize it to understand the narrative passages.

Firstly, we propose to employ external knowledge as the basic resource for restoring the scene. Some previous works (Chen et al., 2019; Guan et al., 2019) also employ external knowledge in Narrative MRC. However, most of them use the concept knowledge from ConceptNet (Liu and Singh, 2004), WordNet (Fellbaum, 1998), or other word-centered knowledge bases to obtain the association information for the noun phrases mentioned in the story. This kind of method is able to help the machine to understand what is mentioned in the story, but not what happens in the story. For example, with those methods, given the sentence “The right side of his face was all covered in blood.”, the machine understand the noun phrases “right side” “face” and “blood” better, but is still unable to know exactly that a man is hurt, he needs medical assistance, and some others nearby might help him. **To this end, we select an event-based knowledge graph, Atomic (Sap et al., 2019), as the source of external knowledge.** Atomic is an atlas of everyday commonsense reasoning. Each center node of Atomic is an event like “PersonX’s face is covered in blood”, and the nodes associated with it are the cause, the effect, and the attribute of the roles of the events. Therefore, Atomic is beneficial for the machine to know “what happens”.

Secondly, we utilize a structured description to restore the scene. **Specifically, we build a scene graph based on the original narrative and the knowledge from Atomic.** Compared with the unstructured text, graph data can represent the

scene more intuitively. In MRC task, previous works (Kipf and Welling, 2016; Qiu et al., 2019) that utilize structured data generally regard the words or noun phrases as the nodes of the graph. Those methods have no specific for Narrative MRC, where the events and the roles are the key factors. Therefore, we build the scene graph by taking the events, the persons, and the external knowledge of the event as the nodes. Meanwhile, we design the connections of the graph from both the perspectives of each event and the whole passage. **Instead of the typical plane graph, we build a three-dimensional graph, which can not only model the relevance among the events in the passage but also retain the unique information of each event.** To encode the graph in a targeted manner, **we propose Graph Dimensional Iteration Network (GDIN).** GDIN can encode the scene graph iteratively and thus obtain the integrated representation of the scene graph. As a result, the machine will understand the narrative more comprehensively and make the decision more precisely.

To summarise, inspired by human behaviors, we propose a novel method to restore the scene for narrative MRC. Specifically, we introduce event knowledge from Atomic (Sap et al., 2019), and build the scene graph to describe the scene. To encode the graph, we propose a novel graph neural network, GDIN. We conduct experiments on two datasets, ROCStories (Mostafazadeh et al., 2017) and CosmosQA (Huang et al., 2019). The results show that our method achieves state-of-the-art.

2 Method

The overview of our method is shown in Figure 2. Our starting point is to let the machine restore the scene like a human while reading narrative passages and then utilize the information from the scene to better comprehension. As shown in Figure 2, given a narrative passage, we firstly obtain

the knowledge for the events mentioned in the passage. Subsequently, we build a scene graph, a three-dimensional graph, whose nodes contain events, person roles, and event knowledge. The graph is composed of two kinds of plane graphs: one is the inner-event graph, which describes a single event; the other is the cross-event graph, which captures the relevance among the events. Meanwhile, we conduct a basic encoding for the narrative passage and the knowledge and then obtain their original representation. By utilizing our proposed Graph Dimensional-Iteration Network (GDIN), we encode the scene graph from the inner-event graph to cross-event graph iteratively. To this end, we obtain the representation of the scene and then make a prediction based on it.

2.1 Knowledge Obtaining

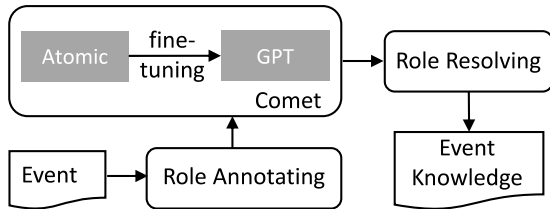


Figure 3: The process of obtaining event knowledge

To endow the model with the ability to associate the event-relevant description, we introduce external knowledge from Atomic. Atomic is an event-based knowledge graph. It contains 24,313 central nodes (i.e., base events) like “PersonX repels PersonY’s attack”. Each of them is linked to multiple types of knowledge nodes, such as the effect on PersonX (e.g., Person X’s heart races), the cause of PersonX (e.g., X wanted to protect himself), the effect on PersonY (e.g., Y gets hurt) and so on. As those knowledge nodes are also events, there are totally 877,108 $\langle event, relation, event \rangle$ triples.

Nevertheless, due to the diversity of real-world events, Atomic cannot cover all the events. Meanwhile, even if the coverage is acceptable for everyday events, the accuracy of event linking (link a certain event text to Atomic) also cannot be ensured. Therefore, we employ the pre-training framework, Comet (Bosselut et al., 2019), which is originally proposed for the task of knowledge base completion. Specifically, Comet is obtained by fine-tuning GPT (Radford et al., 2018) on Atomic. The training task is inputting the start event and the relation

$\langle event, relation, _ \rangle$, and then generating the end event of the triple.

By employing Comet, we design the process of obtaining event knowledge, as shown in Figure 3. Given an event like “Jerry repels Tom’s attack”, to approximate the phrases in Atomic, we firstly annotate the person roles, that is, replacing the subject person with “PersonX” and the other person with “PersonY”. Thus, we get “PersonX repels PersonY’s attack”. Secondly, **we input it to the Comet and obtain the event knowledge**. According to the demand of restoring the scene, we select four types of them, including “xIntend” (Why does X cause the event), “xEffect” (What effects does the event have on X), “yEffect” (What effects does the event have on Y), and “xAttr” (How would X be described). For example, “xIntend” here could be “PersonX wanted to protect himself”. Finally, we resolve the normalized person roles, that is, replacing “PersonX” and “PersonY” with the original person names. For example “xIntend” will finally be “Jerry wanted to protect himself”.

2.2 Scene Graph Building

Having annotated the person roles and obtained relevant knowledge for every event, we build a graph, named “scene graph”, to present a structured description for the scene. We believe that compared with the unstructured text, the graph can provide a more intuitive description from the perspective of the events for the scene.

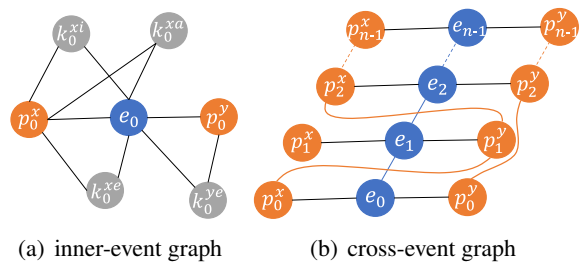


Figure 4: Two types of plane graphs, which compose the three-dimensional scene graph. For i -th ($i \in \{0, 1, 2, \dots, n-1\}$) event, we denote the nodes as follows: e_i (event), p_i^x (PersonX), p_i^y (PersonY), k_i^{xa} (xAttr), k_i^{xi} (xIntend), k_i^{xe} (xEffect), k_i^{ye} (yEffect).

The “Scene Graph Building” part in Figure 2 shows the full view of a scene graph. It is a three-dimensional graph composed of two kinds of plane graphs: inner-event graph and cross-event graph as shown in Figure 4. **The nodes of event and person are the intersection between the two types**

of graphs. The inner-event graph describes a single event, and the cross-event graph captures the relevance among the events, including the narrative order and the person coreference. Accordingly, we build an inner-event graph for each event and a cross-event graph for the whole narrative passage.

The graph contains three kinds of nodes: event, person role, and event knowledge. The links of the graph are designed as follows: (1) In each inner-event graph, a) we link every event knowledge to the event; b) The person roles are linked to the event; c) Each knowledge is linked to its corresponding person. (2) In the cross-event graph, a) we link each event to the adjacent event, which could capture the narrative order; b) To pass information from the perspective of the role, we conduct a coreference resolution and build a connection between two mentions for the same person across the events. To this end, we obtain two kinds of adjacent matrixes for those plane graphs. They are formulated as $A_i^{inner} \in \mathbb{R}^{7 \times 7}$ ($i \in \{0, 1, 2 \dots n-1\}$) and $A^{cross} \in \mathbb{R}^{3n \times 3n}$, where $n-1$ is the total number of the events in the passage.

2.3 Basic Encoding

Before the process of graph encoding, we employ a pre-trained Language Model (LM) to conduct a basic encoding and obtain the original representation of the nodes. For a certain sequence (e.g., a sentence or a passage), the representation is calculated by

$$S^{seq}, S = \mathbf{LM}(\text{sequence}) \in \mathbb{R}^{L_s \times d}, \mathbb{R}^d$$

where L_s denotes the word-level length of the sequence, and d is the dimensional size of the representation. We take S as the sequence representation. Thus inputting the narrative passage to the LM, we can obtain $C^{seq} \in \mathbb{R}^{L_p \times d}$ and $C \in \mathbb{R}^d$. In a specific task, the passage will be concatenated with other text (e.g., question or candidate) together as the input sequence, which will be detailed in 2.5.

In practice, we regard each sentence in the narrative passages as an event, and thus from C^{seq} we can extract $E_i^{seq} \in \mathbb{R}^{L_e \times d}$, the representation of the words of i -th event, according to its sentence span. Then, we merge it by max-pooling, and obtain $E_i^{(0)} \in \mathbb{R}^d$, the original representation of i -th event. Meanwhile, the representation of the roles can be extracted from C^{seq} based on their position as well. Therefore, for the subject person, PersonX, we have $P_i^{x(0)} \in \mathbb{R}^d$; For the other person, Per-

sonY, we have $P_i^{y(0)} \in \mathbb{R}^d$. Specifically, for each role, we take the representation of its first word as its overall representation. Moreover, taking each knowledge as the input of the LM, we can get the representation for it. Hence, for “xIntend”, “xEffect”, “yEffect”, “xAttr”, we have $K_i^{xi(0)}, K_i^{xe(0)}, K_i^{ye(0)}, K_i^{xa(0)} \in \mathbb{R}^d$, respectively.

2.4 Dimensional-Iteration Encoding

To encode the graph in a targeted manner and model the scene from both the perspectives of each event and the whole passage, we propose Graph Dimensional-Iteration Network (GDIN) based on Graph Convolutional Network (GCN) (Kipf and Welling, 2016). As shown in Figure 2, GDIN encodes the graph along the dimension of inner-event graph and then encodes it along the dimension of cross-event graph, which is an iterable process. As the original representation of every node has been obtained by the basic encoding, we conduct a dimensional-iteration encoding with GDIN as follows:

(1) Encoding along the dimension of inner-event graph: At t -step, for i -th inner-event graph, we formulate the representation of its nodes as $H_i^{(t)} = [E_i^{(t)}; P_i^{x(t)}; P_i^{y(t)}; K_i^{xi(t)}; K_i^{xe(t)}; K_i^{ye(t)}; K_i^{xa(t)}] \in \mathbb{R}^{7d}$, where the symbol“;” denotes concatenation. Then we update the representation of all nodes by

$$H_i^{(t+1)} = \sigma \left(D^{in-\frac{1}{2}} A_i^{\tilde{inner}} D^{in-\frac{1}{2}} H^{(t)} W^{in} \right)$$

$$A_i^{\tilde{inner}} = A_i^{inner} + I$$

where I is the identity matrix. $W^{in} \in \mathbb{R}^{7d \times 7d}$ is a trainable matrix and σ is the activation function. $D^{in}_{pp} = \sum_q (A_i^{\tilde{inner}} + I)_{pq}$ is the degree matrix.

(2) Encoding along the dimension of cross-event graph: At $(t+1)$ -step, for the cross-event graph, we collect the nodes of person and event from those above inner-event graphs, and then we formulate the representation of its nodes as $H^{(t+1)} = [E_0^{(t+1)}; P_0^{x(t+1)}; P_0^{y(t+1)}; E_1^{(t+1)}; P_1^{x(t+1)}; P_1^{y(t+1)}; \dots; E_{n-1}^{(t+1)}; P_{n-1}^{x(t+1)}; P_{n-1}^{y(t+1)}] \in \mathbb{R}^{3nd}$. Subsequently, we update the representation of the nodes of person and event by

$$H^{(t+2)} = \sigma \left(D^{cs-\frac{1}{2}} A^{\tilde{cross}} D^{cs-\frac{1}{2}} H^{(t+1)} W^{cs} \right)$$

$$A^{\tilde{cross}} = A^{cross} + I$$

where $W^{cs} \in \mathbb{R}^{3nd \times 3nd}$ is a trainable matrix, and $D_{pp}^{cs} = \sum_q (A^{cross} + I)_{pq}$ is the degree matrix. Note that, in this step the representation of the knowledge does not change. Taking the xEffect knowledge as an example, at this step we have $K_i^{xe(t+2)} = K_i^{xe(t+1)}$.

Iterating: The nodes of event and person are the intersection between the two types of graphs. With iterating (1) and (2), the information passes across different dimensions along those nodes. Therefore, **GDIN can model the three-dimensional scene graph from both the perspectives of each event and the whole passage.** Assuming it iterating for L loops, we obtain $H_i^{(T)} \in \mathbb{R}^{7d}$, where $T = 2L - 1$. The representation of i -th event, $E_i^{(T)} \in \mathbb{R}^d$, can be extracted from $H_i^{(T)}$.

We merge the representation of all the events by $C^s = \sum_i \alpha_i E_i^{(T)}$. The weight α is calculated by

$$\alpha_i = \frac{\exp\left(\sigma\left(\mathbf{w}_p E_i^{(T)}\right)\right)}{\sum_{i'} \exp\left(\sigma\left(\mathbf{w}_p E_{i'}^{(T)}\right)\right)}$$

where $w_p \in \mathbb{R}^d$ is a trainable vector. C^s is the representation of the narrative passage built from the description of the scene. Subsequently, we obtain the final representation of the passage by a residual connection: $C^f = [C^s; C] \in \mathbb{R}^{2d}$.

2.5 Task-Specific Input and Output

We evaluate our method on two types of MRC test, story cloze test and multiple choice. Given a passage, the former requires the model to select a suitable ending from two candidates; the latter requires the model to select the answer for a certain question from four candidates. We prepare the input for the model following Devlin et al. (2018) and Radford et al. (2018). For the story cloze test, we concatenate each ending with the given passage as the input sequence of basic encoding. Then we can obtain an ending-aware passage representation C^{seq} and C . For multiple choice, we concatenate each option with the question and the passage as the input sequence. Thus we get a option-question-aware passage representation C^{seq} and C . After basic encoding and dimensional iteration encoding, we have the final representation C^f . In both the above tests, there is C_j^f , which is the passage representation for j -th candidate. To this end, we score

each candidate by

$$score_j = \frac{\exp\left(C_j^f w_s\right)}{\sum_{j'} \exp\left(C_{j'}^f w_s\right)}$$

where $score_j$ is the normalized selection score of the j -th candidate. $w_s \in \mathbb{R}^{2d}$ is a trainable vector. Then we predict by taking the candidate with the highest score as the ending or the answer.

3 Experiments and Analysis

3.1 Datasets and Metrics

The datasets we choose are ROCStories (Mostafazadeh et al., 2017) and CosmosQA (Huang et al., 2019). The passages of both the above datasets are narrative.

ROCStories: a popular dataset of Story Cloze Test (SCT), annotated by Amazon Mechanical Turk (MTurk) workers based on a collection of short stories. In development and test set, each instance contains a four-sentence passage, and two candidate endings, while the train set only provides the original five-sentence story containing the proper ending. Following previous works (Cai et al., 2017; Chaturvedi et al., 2017; Cui et al., 2019), we take the development set for training and evaluate the performance on the test set.

CosmosQA: a recently proposed dataset formulated as multiple choice. The narratives are collected from the Spinn3r Blog dataset (Burton et al., 2009) and annotated by MTurk. We train and validate the model on the train set and the development set, respectively. As the label of the test set is not public, we evaluate our model by submitting the predictions to the official website¹.

Evaluation Metrics: As the targets of both the above tests are making a choice among the candidates, we use the common metric, accuracy, for evaluation.

3.2 Implementation Details

In practice, we regard each sentence in the narrative passages as an event. When annotating the person roles in a particular sentence, we employ spaCy² for dependency parsing. To link two mentions for the same person across the sentences while building a graph, we utilize Neural Coreference³ for

¹<https://leaderboard.allenai.org/cosmosqa/submissions/public>

²a Python library for natural language processing <https://spacy.io/>

³a toolkit to annotate and resolve coreference clusters <https://github.com/huggingface/neuralcoref>

| Method | Accuracy |
|--------------------------------------|-------------|
| DSSM (Huang et al., 2013) | 58.5 |
| Conditional GAN (Wang et al., 2017a) | 60.9 |
| End Attn (Cai et al., 2017) | 74.7 |
| LR+RNNLM (Schwartz et al., 2017) | 75.2 |
| HCM (Chaturvedi et al., 2017) | 77.6 |
| SeqMANN (Li et al., 2018) | 84.7 |
| GPT-FT (Radford et al., 2018) | 86.5 |
| Concept (Chen et al., 2019) | 87.6 |
| BERT-FT (Devlin et al., 2018) | 89.2 |
| BERT+Diff-Net (Cui et al., 2019) | 90.1 |
| Our method (BERT+GDIN) | 91.9 |

Table 1: Result on ROCStories

coreference resolution. Particularly, in the case where person roles (PersonX or PersonY) could not be found in the sentence, we drop the corresponding nodes (person and relevant knowledge) while building the scene graph.

For a fair comparison with the state-of-the-art models, we employ pre-trained language model, BERT-large (Devlin et al., 2018) and ALBERT-xxlarge (Lan et al., 2020) for basic encoding, respectively. The optimizer we choose is Adam. The learning rates are 5×10^{-6} for the model based on BERT and 1×10^{-5} for that based on ALBERT. We train both the models for three epochs with a 0.1 dropout rate.

3.3 Baselines

We present a series of previous works as baselines for each dataset. For brevity, we only detail those recently published advanced methods.

LM-FT: a kind of model that combines a task-specific output layer with the pre-trained language model, LM. The model is fine-tuned on ROCStories or CosmosQA. LM could be GPT (Radford et al., 2018), BERT, RoBERTa (Liu et al., 2019), or ALBERT. Note that the BERT model is BERT-large, which is the same as that in our method for ROCStories; The ALBERT model is ALBERT-xxlarge and which is the same as that in our method for CosmosQA.

Concept: a neural network model for SCT. This model employs a pre-trained language model, which is initialized from GPT and introduces the external knowledge from ConceptNet.

BERT+Diff-Net: the state-of-the-art model for SCT. It employs the pre-trained language model, BERT (BERT-large). In particular, it focuses on better modeling the differences of each ending and discriminates two endings in three semantic as-

| Method | Accuracy |
|--|-------------|
| Stanford Attentive (Chen et al., 2016) | 44.4 |
| Co-Matching (Wang et al., 2018b) | 44.7 |
| Gated-Attention (Dhingra et al., 2017) | 46.2 |
| Commonsense (Wang et al., 2018a) | 48.2 |
| GPT-FT (Radford et al., 2018) | 54.4 |
| BERT-FT (Devlin et al., 2018) | 67.1 |
| DMCN (Zhang et al., 2020) | 67.6 |
| RoBERTa-FT (Liu et al., 2019) | 80.6 |
| K-Adapter (Wang et al., 2020) | 81.8 |
| ALBERT-FT (Lan et al., 2020) | 82.3 |
| Our method (ALBERT+GDIN) | 84.5 |

Table 2: Result on CosmosQA ⁴

pects: contextual representation, story-aware representation, and discriminative representation.

K-Adapter: a recently proposed advanced method. It contains multiple knowledge-specific adapters. Those adapters infuse entity and syntax knowledge from T-REx (Elsahar et al., 2019) and Book Corpus (Zhu et al., 2015), respectively, into the pre-trained language model (RoBERTa).

3.4 Overall Performance

Table 1 reports the results on the ROCStories dataset. Our proposed method, which restores the scene by the graph and GDIN, outperforms the state-of-the-art model, BERT+Diff-Net (Cui et al., 2019), by 1.9% in terms of accuracy. The results on the CosmosQA dataset are shown in Table 2. Our method outperforms the published state-of-the-art models, K-Adapter (Wang et al., 2020) and ALBERT-FT (Lan et al., 2020), by a considerable margin. Those results demonstrate the effectiveness of our overall method.

3.5 Effectiveness of the Event Knowledge

As stated in 2.1, Knowledge Obtaining, we choose the event relevant knowledge from Atomic instead of the concept knowledge from ConceptNet, WordNet, or other word-centered knowledge bases. To validate the effectiveness of the event knowledge, we employ GPT and RoBERTa for basic encoding. We combine them with GDIN, respectively, and conduct experiments on the two datasets. Table 3 shows the results of the experiments. Our model, GPT+GDIN, surpasses Concept, which utilizes GPT and the knowledge from ConceptNet. Meanwhile, the performance of RoBERTa+GDIN

⁴the published methods by the time of our evaluation submitting (May 16, 2020)

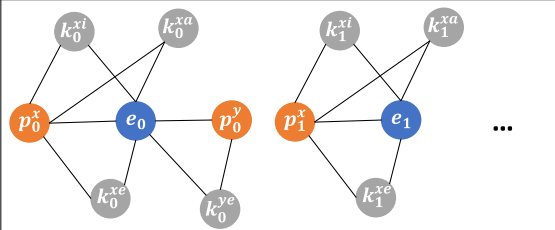
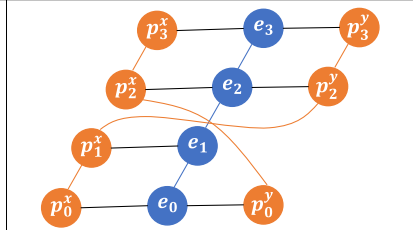
| | | | | | | |
|---|---|--------------------------------------|----------------------------------|---------------------------------------|--|--|
| Narrative Passage | e_0 : Newly married, Susan liked to cook for Bob. | | Person Roles | p_0^x : Susan | p_0^y : Bob | |
| | e_1 : Unfortunately, Susan is a terrible cook. | | | p_1^x : Susan | p_2^y : Susan | |
| | e_2 : Bob ate the food and praise Susan anyway. | | p_2^x : Bob | p_3^y : Susan | | |
| | e_3 : On their first anniversary Bob gave Suan cooking lessons. | | p_3^x : Bob | p_3^y : Susan | | |
| Event Knowledge (roles resolved) | | | | | | |
| k_0^{xa} : Susan is kind | | k_0^{xi} : Susan want to show love | k_0^{xe} : Susan cooks for Bob | k_0^{ye} : Bob feels happy | | |
| k_3^{xa} : Bob is generous | | k_3^{xi} : Bob want to be helpful | k_3^{xe} : Bob is thanked | k_3^{ye} : Susan learns a new skill | | |
| Unstructured Description | | | | | | |
| Susan is kind, and want to show love. Newly married, Susan liked to cook for Bob. Susan cooks for Bob. Bob feels happy. | | | | | | |
| ⋮ | | | | | | |
| Bob is generous, and want to show love. On their first anniversary Bob gave Suan cooking lessons. Bob is thanked. Susan learns a new skill. | | | | | | |
| Scene Graph | | | | | | |
| Inner-Event Graphs |  | | | Cross-Event Graph |  | |
| | Ending 0 : She never cooked again. ✗ | | | | Ending 1 : She became a better cook. ✓ | |

Figure 5: An example for showing the comparison between unstructured description and structured one (graph) for the scene. Note that in the cross-event graph, we omit some links among the persons for brevity, including p_0^x to p_2^y , p_0^x to p_3^y , p_1^x to p_3^y , and p_0^y to p_3^x .

| Method | Accuracy |
|-------------------------------|-------------|
| ROCStories | |
| Concept (Chen et al., 2019) | 87.6 |
| GPT+GDIN | 88.3 |
| CosmosQA | |
| K-Adapter (Wang et al., 2020) | 81.8 |
| RoBERTa+GDIN | 82.5 |

Table 3: Comparison between the different source of knowledge

is better than that of K-Adapter, which employs RoBERTa and entity and syntax knowledge. To a certain extent, those pairs of comparison verify the effectiveness and suitability of the event knowledge we choose for narrative MRC.

3.6 Effectiveness of the Scene Graph

As stated in 2.2, Scene Graph Building, we propose a three-dimensional graph to describe the scene. To verify the advantages of this method, we build two baselines as follows:

BERT+Flat: a method that describes the scene by the flatten unstructured text. Specifically, BERT+Flat attaches the knowledge sentences to their corresponding event text, and organizes an

| Method | Accuracy |
|------------|-------------|
| BERT+Flat | 90.2 |
| BERT+Plane | 90.9 |
| BERT+GDIN | 91.9 |

Table 4: Comparison between different description of the scene

unstructured description by the template:

xAttr + and + xIntend + Event + xEffect + yEffect

where the subject name of xIntend is dropped for fluency. During the process of encoding, the passage joined with the event knowledge is encoded as a whole, and the ending-aware (or option-question-aware) passage representation C is applied directly to predict. Figure 5 shows an example of the comparison between the unstructured description and the structured one for the scene.

BERT+Plane: a method that merges our proposed three-dimensional scene graph into a unified plane graph. Specifically, we put all of the inner-event graphs on a single plane and then build connects among them with the links of the cross-event graph, e.g., the link between e_0 and e_1 . Because GDIN is

| Iteration Step | Accuracy |
|------------------|-------------|
| 1 (no iteration) | 90.8 |
| 2 | 91.9 |
| 3 | 91.4 |
| 4 | 91.0 |
| 5 | 90.9 |

Table 5: Comparison between different iteration steps

not suitable for this plane graph, we encode it by a two-layer GCN instead. The other processes are the same as those in our proposed method.

The comparison results on ROCStories dataset are shown in Table 4. Compared with BERT+Flat, the graph-based method, BERT+GDIN shows significant advantages. The result further confirms our belief that the structured data provides a more intuitive and exploitable description of the scene for the machine. Besides, BERT+GDIN surpasses BERT+Plane, which verifies the effectiveness of our proposed three-dimensional graph. From our point of view, during the process of encoding, the unified plane graph can not retain the unique information of each event as well as the three-dimensional graph does.

3.7 Effectiveness of Iterable Encoding

As stated in 2.4, Dimensional-Iteration Encoding, we propose a novel neural network, GDIN, for encoding the three-dimensional scene graph in a targeted manner. To study the effectiveness of the iteration, we set a different number of iteration steps for our model and conduct experiments on ROCStories. The results are shown in Table 5. On the one hand, when the number is 1, where the model does not iterate actually, the performance lag obviously behind that of 2 steps. This demonstrates the effectiveness of the iteration. On the other hand, by increasing the step number, the performance rises up rapidly and then drops down slowly. This phenomenon indicates that in addition to enabling the iteration, it is also important to select a proper iteration step. We deduce that the proper step is the balance point where each event retains its unique information, and at the same time, also gets the associated information from the whole passage.

4 Related Work

Machine Reading Comprehension: Due to the fast development of deep learning techniques and large-scale datasets, Machine Reading Comprehension(MRC) has gained increasingly wide attention

over the past few years. Richardson et al. (2013) build the multiple-choice dataset MCTest, and this dataset encourages the early research of machine reading comprehension, and a strand of MRC models (Sachan et al., 2015; Narasimhan and Barzilay, 2015) are inspired by the dataset. Hermann et al. (2015) propose a cloze test dataset CNN & Daily Mail, which is large-scale and more suitable than MCTest for deep learning methods. Based on this dataset, Hermann et al. (2015) proposes an attention-based LSTM model named Attentive Reader, and Chen et al. (2016) simplify this model by directly utilize the query-aware context representations to match the candidate answer. Moreover, Rajpurkar et al. (2016) release the span extraction dataset, SQuAD, which has become the most popular MRC dataset over recent years. This dataset enlightens a lot of classical MRC model, like Bidirectional Attention Flow (BiDAF) (Seo et al., 2016) and R-Net (Wang et al., 2017b). Recently, there are some new trends in this field, such as multi-passage MRC (Campos et al., 2016), knowledge-based MRC (Ostermann et al., 2018) and multi-hop MRC (Yang et al., 2018; Min et al., 2019).

Narrative Comprehension: Understanding narrative is a challenging task in natural language understanding, for the passages contain rich cause and effect relations. A large body of previous works focus on scripts learning (Schank and Abelson, 1977). Some previous works addressed script learning by focusing on the narrative cloze test (Chambers and Jurafsky, 2008). Story Cloze Test (Mostafazadeh et al., 2017) is then introduced as a new evaluation framework, and gains wide attention (Chaturvedi et al., 2017; Zhou et al., 2019b). Besides, recent works present other test frameworks for narrative comprehension, such as multiple choice (Huang et al., 2019) and answer generation (Kociský et al., 2018). Compared with the other complex forms of test, e.g., answer generation, the test frameworks we choose (selecting ending or answer) are more focused on narrative comprehension itself.

5 Conclusion

In this paper, we focus on Narrative Machine Reading Comprehension. Inspired by human behaviors, we propose a novel method to restore the scene for the narrative passage. Specifically, we introduce the event knowledge from Atomic and build a three-dimensional graph to describe the scene. To encode the scene graph, we propose Graph

Dimensional-Iteration Network (GDIN). We conduct experiments on two relevant datasets, ROCStories and CosmosQA. The result shows our method achieves state-of-the-art. Further experimental investigation shows that (1) compared with concept knowledge, the event knowledge we choose is more suitable for narrative MRC; (2) Our proposed graph models the scene more effectively than the unstructured text and the unified plane graph do; (3) Our proposed GDIN encodes the scene graph efficiently by iterating multiple steps.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No.61533018, No.61922085, No.61906196) and the Key Research Program of the Chinese Academy of Sciences (Grant NO. ZDBS-SSW-JSC006). This work is also supported by Beijing Academy of Artificial Intelligence (BAAI2019QN0301), the Open Project of Beijing Key Laboratory of Mental Disorders (2019JSJB06) and the independent research project of National Laboratory of Pattern Recognition.

References

- Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. **COMET: Commonsense transformers for automatic knowledge graph construction**. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779, Florence, Italy. Association for Computational Linguistics.
- G. Bower and D. Morrow. 1990. Mental models in narrative comprehension. *Science*, 247 4938:44–8.
- Kevin Burton, Akshay Java, Ian Soboroff, et al. 2009. The icwsm 2009 spinn3r dataset. In *Third Annual Conference on Weblogs and Social Media (ICWSM 2009)*.
- Zheng Cai, Lifu Tu, and Kevin Gimpel. 2017. Pay attention to the ending: Strong neural baselines for the roc story cloze task. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 616–622.
- Daniel Fernando Campos, Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human generated machine reading comprehension dataset. *ArXiv*, abs/1611.09268.
- Nathanael Chambers and Dan Jurafsky. 2008. Unsupervised learning of narrative event chains. In *ACL*.
- Snigdha Chaturvedi, Haoruo Peng, and Dan Roth. 2017. Story comprehension for predicting what happens next. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1603–1614.
- Danqi Chen, Jason Bolton, and Christopher D. Manning. 2016. **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)*, pages 2358–2367, Berlin, Germany. Association for Computational Linguistics.
- Jiao Chen, Jianshu Chen, and Zhou Yu. 2019. Incorporating structured commonsense knowledge in story completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6244–6251.
- Yiming Cui, Wanxiang Che, Wei-Nan Zhang, Ting Liu, Shijin Wang, and Guoping Hu. 2019. Discriminative sentence modeling for story ending prediction. *arXiv preprint arXiv:1912.09008*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Bhuvan Dhingra, Hanxiao Liu, Zhilin Yang, William Cohen, and Ruslan Salakhutdinov. 2017. **Gated-attention readers for text comprehension**. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1832–1846, Vancouver, Canada. Association for Computational Linguistics.
- Hady Elsahar, Pavlos Vougiouklis, Arslan Remaci, Christophe Gravier, Jonathon Hare, Elena Simperl, and Frederique Laforest. 2019. T-rex: A large scale alignment of natural language with knowledge base triples.
- Christiane Fellbaum, editor. 1998. *WordNet: an electronic lexical database*. MIT Press.
- Jian Guan, Yansen Wang, and Minlie Huang. 2019. Story ending generation with incremental encoding and commonsense knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6473–6480.
- Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pages 1693–1701.
- Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos qa: Machine reading comprehension with contextual commonsense reasoning. *arXiv preprint arXiv:1909.00277*.

- Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2333–2338.
- Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
- Tomáš Kočiský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.
- Tomás Kociský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.
- Zhen-Zhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. *ArXiv*, abs/1909.11942.
- Qian Li, Ziwei Li, Jin-Mao Wei, Yanhui Gu, Adam Jatowt, and Zhenglu Yang. 2018. A multi-attention based neural network with external knowledge for story ending predicting task. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1754–1762.
- H Liu and P Singh. 2004. **Conceptnet — a practical commonsense reasoning tool-kit**. *BT Technology Journal*, 22(4):211–226.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Sewon Min, Victor Zhong, Luke Zettlemoyer, and Hananeh Hajishirzi. 2019. Multi-hop reading comprehension through question decomposition and rescoring. In *ACL*.
- Nasrin Mostafazadeh, Michael Roth, Annie Louis, Nathanael Chambers, and James Allen. 2017. Lsdsem 2017 shared task: The story cloze test. In *Proceedings of the 2nd Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics*, pages 46–51.
- Karthik Narasimhan and Regina Barzilay. 2015. Machine comprehension with discourse relations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1253–1262.
- Simon Ostermann, Ashutosh Modi, Michael Roth, Stefan Thater, and Manfred Pinkal. 2018. Mscript: A novel dataset for assessing machine comprehension using script knowledge. *ArXiv*, abs/1803.05223.
- Delai Qiu, Yuanzhe Zhang, Xinwei Feng, Xiangwen Liao, Wenbin Jiang, Yajuan Lyu, Kang Liu, and Jun Zhao. 2019. **Machine reading comprehension using structural knowledge graph-aware network**. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5896–5901, Hong Kong, China. Association for Computational Linguistics.
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/language_understanding_paper.pdf*.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.
- Matthew Richardson, Christopher J.C. Burges, and Erin Renshaw. 2013. **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*, pages 193–203, Seattle, Washington, USA. Association for Computational Linguistics.
- Daniel L. Ruderman and William Bialek. 1993. Statistics of natural images: Scaling in the woods. *Physical review letters*, 73 6:814–817.
- Mrinmaya Sachan, Kumar Dubey, Eric Xing, and Matthew Richardson. 2015. Learning answer-tailing structures for machine comprehension. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 239–249.
- Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. 2019. Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3027–3035.
- Roger C. Schank and Robert P. Abelson. 1977. Scripts, plans, goals and understanding: An inquiry into human knowledge structures.
- Roy Schwartz, Maarten Sap, Ioannis Konstas, Li Zilles, Yejin Choi, and Noah A Smith. 2017. The effect of different writing tasks on linguistic style: A case study of the roc story cloze task. *arXiv preprint arXiv:1702.01841*.

- Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.
- Bingning Wang, Shangmin Guo, Kang Liu, Shizhu He, and Jun Zhao. 2016. Employing external rich knowledge for machine comprehension. In *IJCAI*, pages 2929–2925.
- Bingning Wang, Kang Liu, and Jun Zhao. 2017a. Conditional generative adversarial networks for commonsense machine comprehension. In *IJCAI*.
- Liang Wang, Meng Sun, Wei Zhao, Kewei Shen, and Jingming Liu. 2018a. Yuanfudao at semeval-2018 task 11: Three-way attention and relational knowledge for commonsense machine comprehension. In *SemEval@NAACL-HLT*.
- Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xu-anjing Huang, Jianshu Ji, Cuihong Cao, Daxin Jiang, and Ming Zhou. 2020. K-adapter: Infusing knowledge into pre-trained models with adapters. *ArXiv*, abs/2002.01808.
- Shuohang Wang, Mo Yu, Shiyu Chang, and Jing Jiang. 2018b. A co-matching model for multi-choice reading comprehension. In *ACL*.
- Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017b. [Gated self-matching networks for reading comprehension and question answering](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 189–198, Vancouver, Canada. Association for Computational Linguistics.
- Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. [HotpotQA: A dataset for diverse, explainable multi-hop question answering](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.
- Shuailiang Zhang, Zhao Hai, Yuwei Wu, Zhuosheng Zhang, Xi Zhou, and Xiaoping Zhou. 2020. Dual co-matching network for multi-choice reading comprehension. *ArXiv*, abs/1901.09381.
- Mantong Zhou, Minlie Huang, and Xiaoyan Zhu. 2019a. Story ending selection by finding hints from pairwise candidate endings. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 27(4):719–729.
- Mantong Zhou, Minlie Huang, and Xiaoyan Zhu. 2019b. Story ending selection by finding hints from pairwise candidate endings. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27:719–729.
- Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.
- Rolf A Zwaan, Mark C Langston, and Arthur C Graesser. 1995. The construction of situation models in narrative comprehension: An event-indexing model. *Psychological science*, 6(5):292–297.