C³KG: A Chinese Commonsense Conversation Knowledge Graph

Anonymous ACL submission

Abstract

Existing commonsense knowledge bases often organize tuples in an isolated manner, which is deficient for commonsense conversational models to plan the next steps. To fill the gap, we curate a large-scale multi-turn human-written conversation corpus, and create the first Chinese commonsense conversation knowledge graph which incorporates both social commonsense knowledge and dialog flow information. To show the potential of our graph, we develop a graph-conversation matching approach, and benchmark two graph-grounded conversational tasks. All the resources in this work will be released to foster future research.

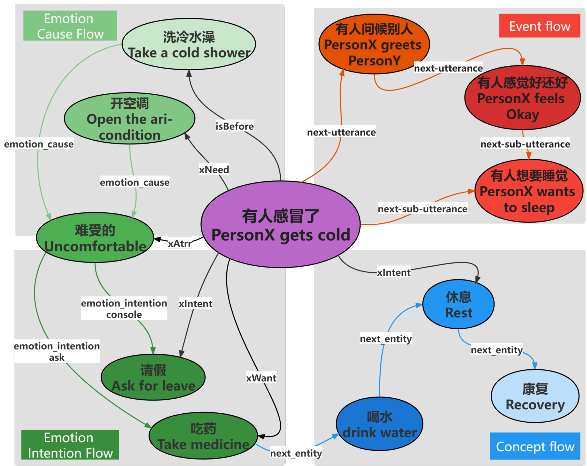

Figure 1: A tiny subset of C³KG, with four unique types of dialog flow relations.

1 Introduction

Commonsense knowledge describes facts and related judgments in our everyday world, which is essential for machine when interacting with humans. These years have witnessed a growing number of literature incorporating commonsense knowledge into various downstream tasks (Bauer et al., 2018; Chen et al., 2019; Lin et al., 2019; Guan et al., 2019; Ji et al., 2020).

Recently, Sap et al. (2019) curate ATOMIC, a large-scale commonsense knowledge base, which covers event-centered social aspects of inferential knowledge tuples. For example, there exist tuples like {*PersonX adopts a cat*, `xEffect`, *happy*} and {*PersonX adopts a cat*, `xWant`, *company*}. Here, `xEffect` and `xWant` are two of nine relations defined in ATOMIC to infer people's mental states for a given event, e.g., *PersonX adopts a cat*. As such, it is promising to detect ATOMIC events mentioned in conversations, and utilize the inferred knowledge when developing social chatbots.

In spite of the potential, it has two major difficulties. For instance, when a friend in *distress* tells us that he recently adopted a cat, we humans will easily suspect that he might has allergies to the cat. However, such reasoning is difficult for chatbots. Given the event-relation pair {*PersonX adopts a cat*, `xEffect`, ___}, ATOMIC contains multiple tails like {*finds out he has allergies*} and the tail {*becomes less lonely*}. To this end, the first difficulty comes from the existence of multiple tails, which will confuse the chatbots when inferring the cause behind the negative emotion. Secondly, the knowledge tuples in ATOMIC are isolated. It is thus more difficult for the chatbots to reason which tail(s) of knowledge should be used to produce coherent responses. For example, if the tuple {*PersonX adopts a cat*, `isAfter`, *finds a cat at the animal shelter*} is detected from the dialogue history, then the tuple {*PersonX adopts a cat*, `xNeed`, *go to an animal rescue center*} should not be considered anymore for future conversations. We argue that these issues hamper the application of ATOMIC to multi-turn dialogue modeling where the conversational agents need not only know the current state but also plan the future **dialog flow**.

To remedy these issues, we define 4 novel dialog flow relations, i.e., event flow, concept flow, emotion-cause flow, emotion-intent flow, as depicted in Figure 1. To build up the relations, we collect a large-scale multi-turn conversations in everyday scenarios, and manually annotate the conversations with emotional information. Based on the annotations, we are able to extract conversation-related events in ATOMIC and connect them using different dialog flows. In this way, we augment ATOMIC with conversation-specific knowledge, which facilitates chatbots to pick out useful commonsense knowledge, and relieves their confusion on noisy knowledge that are incoherent with dialog flows. We believe our graph is favorable for commonsense conversation modeling.

To highlight: (1) We curate a new Chinese corpus, containing multi-turn human-written conversations on dailylife topics and rich, high-quality annotations on the level of sub-utterance; (2) We create and will release the first large-scale **C**hinese **c**ommonsense **c**onversation knowledge graph, **C^{3}KG**, which contain 4 types of unique dialog-flow edges to store the distilled conversation knowledge from the multi-turn conversation corpus; (3) We devise a graph-conversation matching approach, and benchmark 2 typical tasks grounded on commonsense conversation graph.

2 Related Work

2.1 Commmonsense Knowledge Bases

ConceptNet (Speer et al., 2017a) is a popular commonsense knowledge base is, which has a Chinese version with a relatively small set of knowledge (Kuo et al., 2009). Another large-scale commonsense knowledge graph TransOMCS (Zhang et al., 2020) is built automatically by converting syntactic parses of Web sentences into structured knowledge. However, the majority of relations in these knowledge bases are taxonomic relations such as `isA` and `Synonym` (Davis and Marcus, 2015), which inevitably limits their capabilities. Differently, we rely on ATOMIC (Sap et al., 2019). Despite the lack of Chinese version, ATOMIC covers unique mental knowledge. We thus translate it into Chinese and build dialog flow relations on it. Other Chinese knowledge bases include but not limited to CN-DBPedia (Xu et al., 2017) and zhishi.me (Niu et al., 2011).

2.2 Extracting Knowledge from Conversation

To extract structured knowledge from conversations, previous works detect named entities from each utterance in conversational datasets (Xu et al., 2020c; Zou et al., 2021a; Ghosal et al., 2021) and build up the relationship based on their sequential order and Pointwise Mutual Information (PMI) (Church and Hanks, 1990). There also exists some works use automatic extraction tools, such as OpenIE, to construct conversational knowledge bases of certain domains (Ahmad et al., 2020). Although plausible, these knowledge graphs are built on the granularities of word or phrase, which makes them hard to match the overall semantics of dialogue sentences. In this paper, we build a Chinese commonsense conversation knowledge graph based on both multi-turn conversational corpus and event-centered knowledge base. At the same time, we propose to use Sentence-BERT (Reimers and Gurevych, 2019a), a transformer-based semantic similarity model, to construct dialog flow edges in our knowledge graph.

2.3 Knowledge Grounded Dialogue Modeling

There are growing interests in incorporating commonsense knowledge into dialogue tasks. Both Zhou et al. (2018) and Zhang et al. (2019) introduce knowledge triplets from ConceptNet (Speer et al., 2017b) into open-domain response generation. Recently, Li et al. (2021a) and Zhong et al. (2021) exploit ConceptNet to enhance emotion reasoning for response generation, and others design graph reasoning methods to plan the topic transition in the responses (Moon et al., 2019; Tang et al., 2019; Xu et al., 2020a; Li et al., 2021c). One distinct work is Ghosal et al. (2020), which utilizes ATOMIC (Hwang et al., 2020) in emotional dialogue modeling for emotion identification. In this paper, we connect the heads and tails in ATOMIC according to four types of dialog flows. Because the resulted graph C^{3}KG contains both social knowledge from ATOMIC and dialogue knowledge from our corpus, it is thus more suitable for empathetic conversation modeling.

3 A Scenario-based Multi-turn Conversation Corpus

Our aim is to extract common dialog flow information from real conversations. In this way, it is crucial to ensure the quality of the conversation corpus and the reliability of the extraction method. In the following, we firstly introduce the conversation corpus **CConv** we depend on.

Instead using the noisy Internet data, we collect a multi-turn human-written Chinese conversation corpus based on crowdsourcing. Initially, 100 workers are hired, and they are randomly paired to talk in text under a given scenario. Each scenario is one sentence describing the suggested conversation context which often involves certain everyday events. Besides, the workers are also required to follow certain rules like "each utterance should longer than 6 Chinese characters", which are critical to help ensure the quality of the collected conversation. At the beginning of the crowdsourcing, we check each collected conversation and re-train the workers. To ensure the quality, we keep only 62 well-trained workers and let them finish our task. Note that the workers are paid with 1 CNY per utterance (nearly 0.2 dollar per utterance). Finally, we obtain 32k sessions of high-quality two-party conversations (650k utterances in total) on 200 scenarios of 15 daily topics.

To facilitate future research, we then hire another 3 well-trained assistants to manually annotate the conversations with fine-grained emotional labels including speaker's emotion type, emotion cause, and response intention type. Following Rashkin et al. (2019), we define emotion type with 5 general classes {joy, angry, sad, caring, other}. Emotion cause span is a continuous text spans implying the reason of certain emotion (Li et al., 2021b). Response intention type is essential for building empathetic chatbots, and we define 6 commonly-adopted intent classes of {ask, advise, describe, opinion, console, other} following Welivita and Pu (2020). A snippet of an conversation example is given in Figure 2. In Appendix, we present more information of the constructed corpus.

By utilizing the annotations, we are able to distill dialogue knowledge to enhance the conversation graph and graph-grounded conversation modeling.

4 Translation of ATOMIC

Because our conversation corpus is Chinese, we want to build a Chinese conversation knowledge graph. It is well known that to build a knowledge graph from scratch is laborious and time-consuming. Instead, we base on ATOMIC and design a pipeline method to translate it into Chinese, meanwhile ensuring the resulted knowledge graph is reliable and suitable for conversation grounding.

4.1 Brief Introduction of ATOMIC

Before describing our detailed processing steps, we firstly give a brief description of ATOMIC (Sap et al., 2019). ATOMIC organizes commonsense knowledge in the form of triplet <head, relation, tail>, where head often describes a daily event.

There are two unique properties making ATOMIC suitable and attractive for building empathic chatbots. Firstly, ATOMIC collects knowledge about how people will react to a given event. This kind of knowledge is related to people's mental states, which is beneficial for understanding implicit emotions. For example, given a head event *PersonX makes PersonY's coffee*, ATOMIC contains knowledge that PersonY will be *grateful* along the relation `oReact`. Secondly, ATOMIC organizes knowledge using several inferential relations and naturally supports *if-then* reasoning, which is crucial generating coherent responses. Totally, there are 9 relations defined in ATOMIC. The details can be found in Appendix.

4.2 Replacement of Certain Tokens

We begin with translating high-frequency patterns in the original triplets. As compared to the predefined set of relations, it is more difficult to handle the heads and tails. In ATOMIC, for example, there exist **185,046** heads and tails containing tokens like "*PersonX*" and "*PersonY*". These personal pronouns stand for the givers and the receives for a certain event, and can be regarded as the speech parties in a conversation. Also, some ATOMIC heads like {*PersonX gets ____ as a pet*}, have a blank which can be filled with various tokens.

These aforementioned patterns bring ambiguity to the triplet semantics, and will confuse the translation model. To address, we devise a series of replacement rules to keep the original semantics while translation. For example, for the ATOMIC head *PersonX votes for personY*, we convert it to be "Someone votes for someone else" and send it to our translation model.

4.3 Joint Translation of Head and Tail

Nevertheless, the majority of the heads and tails in ATOMIC are short phrases, while machine translation models are often context-based. The multi-sense characteristics of language will further deteriorate the translation quality if we separately feed each single head and tail to a translation model.

To remedy the issues, we instead translate the head and tail in each triplet together. Given a triplet $<h, r, t>$, we connect the head h with its t using a heuristic connecting word r' w.r.t. the relation r, and obtain one long sentence l. After translating

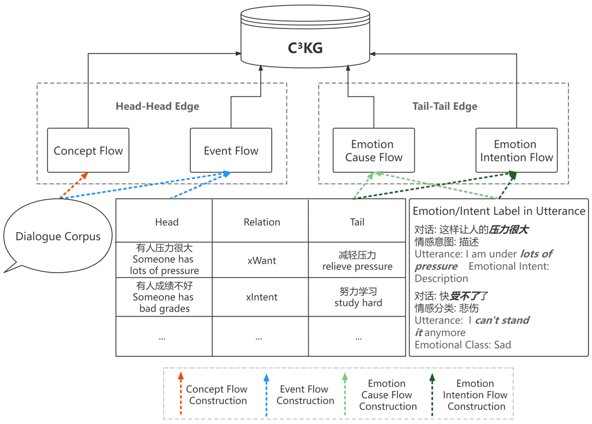

Figure 2: Construction Process of C^{3}KG.

the long text, we split the translation result with the connecting word and turn it into h_{tr} and t_{tr}:

$$\begin{aligned} l &= \mathrm{CONNECT}(h, r', t) \\ l'_{tr} &= \mathrm{TRANSLATION}(l) \\ h_{tr}, r_{tr}, t_{tr} &= \mathrm{SPLIT}(m'_{tr}, r'_{tr}) \end{aligned} \tag{1}$$

where the resulted <h_{tr}, r_{tr}, t_{tr}> is the translated triplets. And CONNECT, SPLIT denote the corresponding operation. TRANSLATION stands for our translation model. By this means, we expect the connected l provides more contextual information for better semantic translation. The comparison results between separate translation and joint translation will be given in Section 6.3.1.

Note that auxiliary translation methods can be used. In this work, we use Xiaomi commercial Translation service.[1] For simplicity, we denote the translated ATOMIC as **ATOMIC-zh**.

5 Conversation Knowledge Graph Construction

5.1 Overview of C^{3}KG

To supply dialog flow information for commonsense reasoning, we create a **C**hinese **C**ommonsense **C**onversation **K**nowledge **G**raph, **C^{3}KG**, whose statistics are summarized in below.

We then introduce our method of constructing a conversational knowledge graph based on ATOMIC-zh and our multi-turn conversation corpus. In general, we extract events from each conversations and match with the head in ATOMIC-zh. The core is how to build new dialog flow relations, which is depicted in Figure 2, and will be detailed present in the following section.

[1] http://fanyi.mioffice.cn

#Relations	ATOMIC Relations	63,6656
	Event Flows	57,1196
	Concept Flows	7,7587
	Emotion-Cause Flows	187
	Emotion-Intent Flows	196
#Triplets	1285,822	

Table 1: Statistics of C^{3}KG.

5.2 Event Extraction

Knowledge in ATOMIC-zh is event-based and most of them are declarative sentences with some entities omitted. However, utterances in the open-domain dialogue dataset contain a lot of colloquial expressions and sub-sentences with more complex structures. To cope with the complexity, we develop a dependency parsing-based event detection pipeline to extract salient events in each utterance. The overview of our algorithm is described in Algorithm 1.

Pre-processing. We first split each utterance with punctuation, and operate on the level of sub-utterances. To reduce noise, we then filter short sub-utterances with transitive and dumb semantics like "好的" (OK), "就是这样" (That's it). After that, we perform Dependency Syntactic Parsing and POS tagging using ltp4[2], and extract event mentions based on two kinds of structural patterns, verb-driven and adjective-driven clauses.

Verb-driven. Verb-driven clauses have a verb connecting to the root node in the dependency tree. After filtering some noisy words, we obtain verb-driven event mentions. For example, we extract the mention "催促提供物资的商家" (urged the merchants who provide supplies) from utterance "我和上司已经在催促提供物资的商家了" (My boss and I have already urged the merchants who provide supplies). In this utterance, we filter subject of utterance"我和上司" (My boss and I), adverbial"已经" (have already) and modal particle"了" (yet) at the end of the utterance.

Adjective-driven. Besides, adjective-driven clauses often have meaningful entities in sub-utterances. Similarly, we extract adjective-driven event mentions based on the adjective-driven clauses by keeping the modifier of its key adjective and filtering out other words. For example, we extract the mention "学习节奏快" (The pace of learning is fast) from the utterance "但学习节奏也太快了吧" (But the pace of learning is too fast).

[2] https://github.com/HIT-SCIR/ltp

Algorithm 1 Event Extraction from Utterance

Input: An utterance U

Output: A set of event mentions M

Split U with punctuation, and get a series of sub-utterance SU, filter SU based on length
for each $su \in SU$ **do**
  Obtain the dependency tree dep and POS tagging result pos of su
  Find the had node which connects directly to the $ROOT$ node in the dependency tree
  if POS tag of the had node $\in [v, a]$ **then**
    Append had to HAD
  end if
  if The number of verbs connected directly to had more than 1 **then**
    Recursively search verbs in the sub-tree of had and replace had in HAD with the founded verbs
  end if
  for $had \in HAD$ **do**
    if POS of node had is v **then**
      Keep words in su that appear after had and words connect directly to had and relation is 'ADV', connect them and append to M
    else
      Remain words in su that connect directly to had and relation is 'SBV', connect them and append to M
    end if
  end for
end for
Return M

In this utterance, we filter the initial conjunction "但是" (but), adverbial "也" (no meaning) and "太" (too) and modal particle "了" (yet) and "吧" (no meaning) at the end of the utterance.

Recursive Applying. The resulted event mentions may still contain multiple verbs and several semantic units. In this case, we apply a secondary decomposition. For example, we will split the event mention "以为进了大学就可以放松放松" (could relax after entering university) into two events "进了大学" (entering university) and "就可以放松放松" (could relax). To do so, we count the number of verbs connected to the root word in the mention as well as the depth of the sub-trees led by those verbs. Based on the results, we determine whether the mention needs a secondary decomposition using a threshold. If needed, we recursively search verbs in the original dependency tree and replace the key verb with the verbs we found.

5.3 Event Linking as Matching

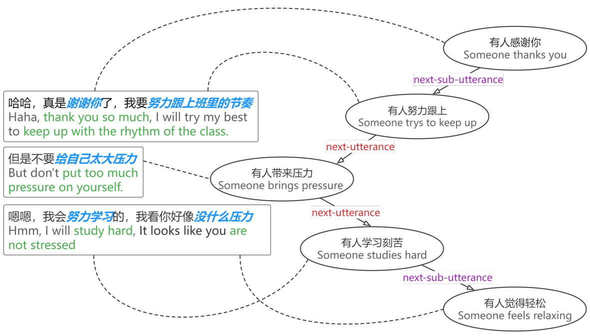

Figure 3: An Example of Head-Head Edge Construction for Event Flows.

In order to discover common dialog flows among the knowledge base, the event mentions in the conversations are then linked to ATOMIC heads using matching techniques.

Typically, we adopt Sentence-BERT, a powerful semantic matching model, which is based on Siamese and Triplet Network and pre-trained on sentence pairs in different relationships (Reimers and Gurevych, 2019b). It encodes two given sentences separately and calculates the similarity between their representations, and thus performing efficiently in large-scale many-to-many matching.

To enhance the matching performance, we fine-tune Sentence-BERT on our corpus. Specifically, we randomly select 8,000 <m, h> mention-head pairs matched by pre-trained Sentence-BERT, and manually label a matching score in $\{0,1\}$ for fine-tuning. Note the reason why we adopt discrete $\{0,1\}$ instead of continuous $[0, 1]$ scores is that using the former effectively mitigates the domain gap. It will induce the matching model to label 0 for those <m, h> share similar characters in surface but different meanings in semantics. After fine-tuning, we calculate the cosine similarity scores and choose the head with the highest score as the matching result given an event mention.

5.4 Edge Construction

Now we have 32k sessions of multi-turn conversations and link their event mentions to ATOMIC heads. The remaining is how to utilize them and build commonsense conversation knowledge graph.

In this work, we propose three kinds of edges to reflect different types of dialog flows.

5.4.1 Head-Head Edge Construction

Event Flow. Naturally, a dialogue is hierarchical in that it consists of a sequence of utterances produced by two interlocutors, where each utterance is composed of one or several sub-utterances. If two event mentions are detected together within in a conversation, the co-occurrence can be regarded as a dialog flow example. Following the flow, it is then intuitive to connect the ATOMIC heads linked by the mentions, as illustrated in Figure 3. By connecting intra-utterance and inter- utterance mentions, we acquire the event flows of `next-sub-utterance` and `next-utterance`.

Concept Flow. ATOMIC also has entity-level heads in addition to the phrase-level events. To utilize them, we perform entity linking by detecting word entities with POS tag belonging to {verb, noun, adjective} in the original conversations, and match them with the entity-level AOMIC heads to construct concept flow edges similarly. These concept flows are helpful for planning and transiting the contents in topic-aware conversation (Yao et al., 2018; Moon et al., 2019; Xu et al., 2020b; Zou et al., 2021b).

Because we are interested in the most common dialog flows, we only keep those highly-frequent connections, and create a head-to-head dialog flow between the ATOMIC head entities and events.

5.4.2 Tail-Tail Edge Construction

Besides, we also consider another essential type of dialog flow, i.e., emotion-based empathy flow. In this paper, we utilize the emotional labels on our corpus (in Section 3) to construct two kinds of emotion-based edges connecting tails in our knowledge graph. Intuitively, `emotion-cause` dialog flow reflects the reasons for a specific emotion, which is useful for fine-grained emotion understanding. And `emotion-intent` empathy flow indicates what response intentions are proper to use when the other one is in a specific emotion, which is critical for response empathy.

Pre-processing. To construct emotion-based edges, we category the tails into 3 classes according to their connecting relations, as listed in Table 2. The first class of tails are linked by relations $xAttr$ or $xReact$, which reflects people's psychological reaction towards a certain event (head). For instance, {*PersonX runs out of steam*, `xAttr`, *tired*} indicates that someone is lacking energy. We denote the first class as Tail$_{emotion}$. The second class Tail$_{before}$ states the events commonly happen before the heads, e.g., {*PersonX runs out of steam*, `isAfter`, *PersonX exercises in gym*}. On the contrary, the last class Tail$_{after}$ contain the events following the head events like {*PersonX runs out of steam*, `xWant`, *to get some energy*}.

Tail$_{emotion}$	xAttr,xReact
Tail$_{before}$	isAfter, xNeed
Tail$_{after}$	isBefore, xWant, xIntent, xEffect, oEffect

Table 2: Relation Categories For Emotion-based Edge Construction.

By analyzing these relations and tails, we find heuristics to build emotion-based dialog flows. By connecting the head and tails in class Tail$_{emotion}$, we are able to create causal emotional inference like {*PersonX exercises in gym*, `emotion-cause`, *tired*}. Through cross linking the tails in class Tail$_{emotion}$ and Tail$_{after}$, we are able to develop the inferential edges like {*tired*, `emotion-intent`, *to get some energy*}.

Filtering. Based on the heuristics, we apply Sentence-BERT[3] to match each tail in class Tail$_{emotion}$ to one of 4 emotion labels defined in our dataset, i.e., {joy, sad, angry, caring}, and set a threshold of 0.7 to determine the emotion class of the tails. The tails sharing the same emotion class with the original utterance are kept to build emotion-based dialog flows.

Emotion Cause Flow. Then, we apply keyword-based exact matching between the tails in Tail$_{before}$ with dialogue context. For Tail$_{before}$, if there is an keyword exactly matched with some keywords in the previous utterances, we create an $emotion-cause$ edge flowed from the tail of Tail$_{before}$ to those filtered tails in Tail$_{emotion}$, indicating that the event of Tail$_{before}$ may cause person to feel the emotion of the tail in Tail$_{emotion}$.

Emotion Intention Flow. For tails in class Tail$_{after}$, we create an `emotion_intent` flow from those filtered tails in Tail$_{emotion}$ to the tails in Tail$_{after}$. Notably, we also assign one of five intent labels to each `emotion_intent` edge, i.e., {ask, advise, describe, opinion, console} (Section 3).

Figure 4 depicts the process of constructing the labeled `emotion-intent` edge. We start by

[3]This model is not fine-tuned on our dataset.

matching the tail *Uncomfortable* in $\text{Tail}_{emotion}$ to the utterance emotion label "sad". Then, we detect that the tail *Take medicine* in Tail_{after} shows up in the next utterance. As such, we build a `emotion_intent` edge from the tail *Uncomfortable* to tail *Take medicine*, and add the intent label of the second utterance "ask" on to the edge. This kind of tail-tail `emotion_intent` flows is supportive for chatbots to choose proper response strategy under a certain situation.

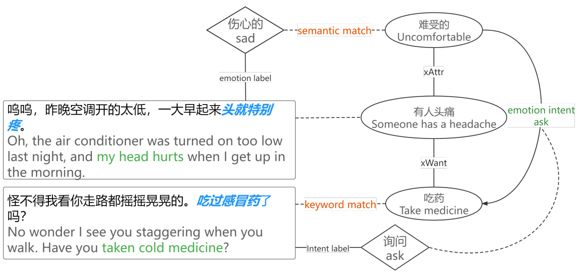

Figure 4: An Example of Tail-Tail Edge Construction for Emotional Flows.

Method	SBERT		SBERT-finetune	
	Similarity	Number	Similarity	Number
Simple	0.513	**1.57**	0.532	**1.57**
POS	**0.514**	0.75	0.541	0.75
Parsing	0.513	1.53	**0.553**	1.53

Table 3: Comparison of Matching Approaches.

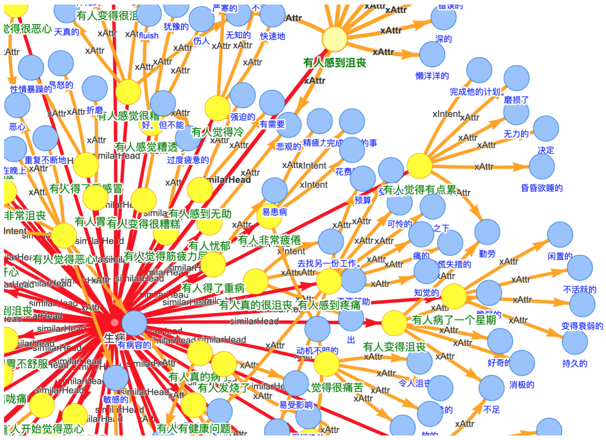

Figure 5: Scenario Graph of "Sickness".

6 Evaluation

6.1 Matching Evaluation

We randomly choose 100 utterances to evaluate our event extraction (Section 5.2) and matching methods (Section 5.3). We denote our proposed method as *Parsing*. To compare with it, we use another two methods to process utterances: *POS* employs POS tagging-based templates to extract events, and *Simple* only splits and filters utterances according to punctuation before matching. We report matching results using both Sentence-BERT and Sentence-BERT-finetune.

In Table 3, Similarity stands for the averaged matching degree, and Number for the average number of matched ATOMIC heads of the chosen utterances, which can be seen as an indicator for matching recall. Although the three methods have similar average similarity without finetuning, our Parsing method gets an obvious 0.4 similarity improvement after finetuning as compared with Simple and POS without loss of knowledge recall, which is also significantly better than POS-based method.

6.2 Scenario Graph

To fully utilize our scenario description, we also build a scenario-based knowledge graph, using the matching approach proposed in this work. By quantifying the relevance of and visualizing the matched result for each topic of scenarios, we are able to better understand the matching quality.

Specifically, we use sub-sentence to match heads in ATOMIC-zh, and use the top 0.5% heads we match in each scenario to build scenario-based graphs. Each of them can be seen as a sampled sub-graph from ATOMIC-zh, with higher topic coherence with its scenario. After annotation, the matching accuracy based on 3 annotators reaches 0.71, which indicate a fair quality of scenario graph. To depict, we visualize a snippet of the scenario graph "sickness" in Figure 5. Please kindly note that for clarity, we only visualize a small set of relation and tails in Figure 5. In fact, every scenario graphs contain the full set of C^3KG relations. For more scenario graphs, please check Appendix.

6.3 Graph Evaluation

6.3.1 Node Evaluation

Since our C^3KG is built upon the translated ATOMIC-zh. We firstly evaluate the quality of our graph in terms of translation accuracy. In specific, we randomly sample 200 triplets from C^3KG, and ask annotators to label each Chinese triplet in terms of fluency and logic correctness with $\{0,1\}$ scores. To validate our joint translation method, we also compare with the results using separate translation.

As shown in Table 4, the significant increases

Method	Fluency	Logic
Separate translation	0.825	0.71
Joint translation	**0.92**	**0.88**

Table 4: Evaluation of Translation Quality.

on both Fluency and Logic aspects clearly demonstrate the superiority of joint translation method. In terms of logical coherence, we find many sample cases are labeled with 0 logical score due to the incompleteness of their heads, which somehow confuses the semantics and obstacles logical connection to the tails. For example, {有人把他父亲, `xAttr`, 告密者} ({*PersonX gets PersonX's father*, `xAttr`, *a tattletale*}) seems ridiculous. However, if we add 出卖 (*betrayed*) in the end of the heads, then it becomes complete and logical. Nonetheless, such seemingly illogical knowledge might still be informative for downstream tasks with fuzzy matching techniques. Hence, we retain this kind of incomplete heads.

6.3.2 Edge Evaluation

At the heart of C^{3}KG is the novel dialog flow relations we develop in this work. To validate the quality of these relations, we utilize another open-domain multi-turn Chinese dialogue dataset, MOD (Fei et al., 2021)[4]. In specific, we extract event mentions from MOD utterances and match them to our graph using the methods as in Section 5.2. Then we evaluate the connectivity and distance of the matched results, w.r.t. both `next_utterance` and `next_sub_utterance` relations. This aims to assess the aggregation degree of related content in our knowledge graph.

KG	next_utterance		next_sub_utterance	
	Connectivity	Dist.	Connectivity	Dist.
C^{3}KG	45.50	2.08	12.96	2.13

Table 5: Edge Evaluation Result on MOD dataset.

Table 5 shows our edge evaluation result on MOD. While the connectivity of the matching node between utterance is quite high, we find that the connectivity of matching node within the same utterance is relative fair. This result gives us inspiration in the future study to enlarge window size to find more latent event-level transfer within the utterance.

[4] https://github.com/lizekang/DSTC10-MOD

Method	Emotion	Intent
Base	90.7	65.3
Knowledge	**91.4**	**67.3**
History	90.5	64.7
Knowledge+History	91.2	65.4

Table 6: Baselines for Graph-grounded Tasks.

7 Proposed Tasks

To show the potential, we propose two graph-grounded conversational tasks, i.e., emotion classification and intent prediction, and train benchmark models using our labeled corpus CConv.

Task 1: Emotion Classification requires to produce an emotion label conditions on the conversations. Following common practice, we choose the BERT model, and sample the `xAttr`, `xReact` tails from our matching head as extra input.

Task 2: Intent Prediction requires to predict a proper type of response intent for the conversations. We choose BERT model, and sample the `oReact`, `oEffect` tails from our matching heads. As simple baselines, we introduce history and graph knowledge through concatenation with an input format as U_{i-2} [SEP] U_{i-1} [SEP] U_i [SEP] `OReact` tail [SEP] `oEffect` tail.

The accuracies of baseline methods are reported in Table 6. $Base$ denotes only using the utterance to do prediction. $Knowledge$ and $History$ denote whether to add knowledge we sampled and dialogue history to the model. While adding knowledge improves the emotion classification and intent prediction performances, it seems problematic to directly concatenating history dialogues, which may bring noises. The moderate scores also indicate that there is still a room to improve for graph-grounded conversation understanding.

8 Discussions of Future Work

In this work, we provide a systematic approach from event mention detection, event linking to conversation graph construction which consists of 4 distinguished types of dialog flows. For each step, there exist possible refinements. For example, we plan to include other event-based resources to improve graph-conversation matching accuracy as well as the graph knowledge coverage.

We also plan to continue the annotations to supply more dialog flow information especially those empathy ones, and evaluate more dialog flow relations on other datasets. Ethical statements are given in Appendix.

References

Zishan Ahmad, Asif Ekbal, Shubhashis Sengupta, Anutosh Mitra, Roshni Rammani, and Pushpak Bhattacharyya. 2020. Active learning based relation classification for knowledge graph construction from conversation data. In *International Conference on Neural Information Processing*, pages 617–625. Springer.

Lisa Bauer, Yicheng Wang, and Mohit Bansal. 2018. Commonsense for generative multi-hop question answering tasks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4220–4230, Brussels, Belgium. Association for Computational Linguistics.

Jiaao Chen, Jianshu Chen, and Zhou Yu. 2019. Incorporating structured commonsense knowledge in story completion. In *AAAI*.

Kenneth Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29.

E. Davis and G. Marcus. 2015. Commonsense reasoning and commonsense knowledge in artificial intelligence. *Communications of the ACM*, 58:92 – 103.

Zhengcong Fei, Zekang Li, Jinchao Zhang, Yang Feng, and Jie Zhou. 2021. Towards expressive communication with internet memes: A new multimodal conversation dataset and benchmark. *arXiv preprint arXiv:2109.01839*.

Deepanway Ghosal, Pengfei Hong, Siqi Shen, Navonil Majumder, Rada Mihalcea, and Soujanya Poria. 2021. Cider: Commonsense inference for dialogue explanation and reasoning. *arXiv preprint arXiv:2106.00510*.

Deepanway Ghosal, Navonil Majumder, Alexander Gelbukh, Rada Mihalcea, and Soujanya Poria. 2020. Cosmic: Commonsense knowledge for emotion identification in conversations. *arXiv preprint arXiv:2010.02795*.

Jian Guan, Yansen Wang, and Minlie Huang. 2019. Story ending generation with incremental encoding and commonsense knowledge. In *AAAI*.

Jena D Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. 2020. Comet-atomic 2020: On symbolic and neural commonsense knowledge graphs. *arXiv preprint arXiv:2010.05953*.

Haozhe Ji, Pei Ke, Shaohan Huang, Furu Wei, Xiaoyan Zhu, and Minlie Huang. 2020. Language generation with multi-hop reasoning on commonsense knowledge graph. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 725–736, Online. Association for Computational Linguistics.

Yen-Ling Kuo, Jong-Chuan Lee, Kai yang Chiang, Rex Wang, Edward Shen, Cheng wei Chan, and Jane Yung jen Hsu. 2009. Community-based game design: experiments on social games for commonsense data collection. In *HCOMP '09*.

Qintong Li, Piji Li, Zhumin Chen, and Zhaochun Ren. 2021a. Towards empathetic dialogue generation over multi-type knowledge. In *AAAI*.

Yanran Li, Ke Li, Hongke Ning, Xiaoqiang Xia, Yalong Guo, Chen Wei, Jianwei Cui, and Bin Wang. 2021b. Towards an online empathetic chatbot with emotion causes. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2041–2045.

Yanran Li, Wenjie Li, and Zhitao Wang. 2021c. Graph-structured context understanding for knowledge-grounded response generation. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. Kagnet: Knowledge-aware graph networks for commonsense reasoning. *ArXiv*, abs/1909.02151.

Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. 2019. Opendialkg: Explainable conversational reasoning with attention-based walks over knowledge graphs. In *ACL*.

Xing Niu, Xinruo Sun, Haofen Wang, Shunlin Rong, Guilin Qi, and Yong Yu. 2011. Zhishi.me - weaving chinese linking open data. In *SEMWEB*.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *ACL*.

Nils Reimers and Iryna Gurevych. 2019a. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Nils Reimers and Iryna Gurevych. 2019b. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

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 *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 3027–3035. AAAI Press.

R. Speer, Joshua Chin, and Catherine Havasi. 2017a. Conceptnet 5.5: An open multilingual graph of general knowledge. In *AAAI*.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017b. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-first AAAI conference on artificial intelligence*.

Jianheng Tang, Tiancheng Zhao, Chenyan Xiong, Xiaodan Liang, Eric P. Xing, and Zhiting Hu. 2019. Target-guided open-domain conversation. *ArXiv*, abs/1905.11553.

A. Welivita and Pearl Pu. 2020. A taxonomy of empathetic response intents in human social conversations. In *COLING*.

Bo Xu, Yong Xu, Jiaqing Liang, Chenhao Xie, Bin Liang, Wanyun Cui, and Yanghua Xiao. 2017. Cndbpedia: A never-ending chinese knowledge extraction system. In *IEA/AIE*.

J. Xu, Haifeng Wang, Zheng-Yu Niu, Hua Wu, and Wanxiang Che. 2020a. Knowledge graph grounded goal planning for open-domain conversation generation. In *AAAI*.

J. Xu, Haifeng Wang, Zheng-Yu Niu, Hua Wu, Wanxiang Che, and Ting Liu. 2020b. Conversational graph grounded policy learning for open-domain conversation generation. In *ACL*.

Jun Xu, Haifeng Wang, Zheng-Yu Niu, Hua Wu, Wanxiang Che, and Ting Liu. 2020c. Conversational graph grounded policy learning for open-domain conversation generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1835–1845.

Lili Yao, Ruijian Xu, C. Li, Dongyan Zhao, and Rui Yan. 2018. Chat more if you like: Dynamic cue words planning to flow longer conversations. *ArXiv*, abs/1811.07631.

Hongming Zhang, Daniel Khashabi, Yangqiu Song, and Dan Roth. 2020. Transomcs: From linguistic graphs to commonsense knowledge. In *IJCAI*.

Houyu Zhang, Zhenghao Liu, Chenyan Xiong, and Zhiyuan Liu. 2019. Grounded conversation generation as guided traverses in commonsense knowledge graphs. *arXiv preprint arXiv:1911.02707*.

Peixiang Zhong, Di Wang, Pengfei Li, Chen Zhang, Hao Wang, and Chunyan Miao. 2021. CARE: commonsense-aware emotional response generation with latent concepts. In *AAAI*.

Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. 2018. Commonsense knowledge aware conversation generation with graph attention. In *IJCAI*, pages 4623–4629.

Yicheng Zou, Zhihua Liu, Xingwu Hu, and Qi Zhang. 2021a. Thinking clearly, talking fast: Concept-guided non-autoregressive generation for open-domain dialogue systems. *arXiv preprint arXiv:2109.04084*.

Yicheng Zou, Zhihua Liu, Xingwu Hu, and Qi Zhang. 2021b. Thinking clearly, talking fast: Concept-guided non-autoregressive generation for open-domain dialogue systems. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2215–2226, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

A Ethical Considerations

At last, we discuss the potential ethic impacts of this work. (1) **Transparency**: We will release the newly introduced corpus and the built conversation knowledge graph, as well as the benchmark approaches to facilitate future research. Similar datasets and knowledge bases include EmpatheticDialogues (Rashkin et al., 2019) and ATOMIC (Sap et al., 2019), which are often public available and have been used extensively. (2) **Privacy**: The corpus is crowdsourced under a set of specific rules to forbid the workers disclosure sensitive and personal identifiable information. (3) **Politeness**: Because our conversations are human-written and are related to healthy dailylife scenarios, they are expected to be clean, legal, and polite. The crowdsourcing rules are designed to avoid emotionally triggering words as much as possible.

B Corpus: CConv

B.1 Example & Statistics

In our corpus CConv, conversations are conducted based on a scenario between two parties. Table 8 gives an example conversation. The statistics of CConv is also present in Table 7. Since there are 200 scenarios in total, and hence we have 160 diverse multi-turn conversations in average.

# sessions of dialogues	32,612
# utterances	650,147
# unique scenarios	200
# conversation topics	15
Avg. # words per utterance	7.8
Avg. # turns per dialogue	19.9

Table 7: The Statistics of the Corpus CConv.

B.2 Topics and Scenarios

To ensure the diversity of the conversations, we select 15 everyday topics. For each topic, we manually write tens of one-sentence scenario to guide the conversation context.

In total, we have 15 topics and 200 scenarios. To better understand, we show some example topics and scenarios in Table 9.

B.3 Annotation Criteria

To facilitate future research, we hire another 3 well-trained assistants to manually annotate the conversations with fine-grained emotional labels including speaker's emotion type, emotion cause, and response intention type. The annotation example in given along with the example in Table 8.

Emotion Class. Following Rashkin et al. (2019), we define emotion type with 5 general classes {joy, angry, sad, caring, other}.

Emotion Cause Span. Emotion cause span is a continuous text spans implying the reason of certain emotion (Li et al., 2021b).

Response Intent. Response intention type is essential for building empathetic chatbots, and we define 6 commonly-adopted intent classes of {ask, advise, describe, opinion, console, other} following Welivita and Pu (2020), which are described in Table 10.

C ATOMIC

In this work, we introduce ATOMIC (Sap et al., 2019) as the commonsense knowledge base due to its attractive properties of mental state inferences and *if-then* causal relations, as analyzed before.

ATOMIC (Sap et al., 2019) is a novel event-centered knowledge graph, consisting of 880K tuples of social commonsense knowledge. Distinguished from ConceptNet (Speer et al., 2017a), there are two unique properties making ATOMIC suitable and attractive for building empathic chatbots. Firstly, ATOMIC collects knowledge about how people will feel and react to a given event. This kind of knowledge is related to people's mental states, which is beneficial for understanding implicit emotions. For example, given a head event *PersonX makes PersonY's coffee*, ATOMIC contains knowledge that PersonY will be *grateful* along the relation `oReact`. Secondly, ATOMIC organizes knowledge using several inferential relations and naturally supports *if-then* reasoning, which is crucial generating coherent responses.

Here, we adopt the figures and demonstrations from the original ATOMIC paper (Sap et al., 2019) to present the 9 relations defined in ATOMIC and give some examples in Figure 6 and Table 11.

D Evaluation

D.1 Template-based Event Extraction Methods

To evaluate our matching methods proposed in this work, we randomly choose 100 utterances and compare with several approaches. In specific, we propose a baseline POS matching method, which employs POS tagging-based templates to extract events. The templates are given in Table 12.

Situation			
同事之间，一方身体不舒服，另一方表达关心 Acted as colleagues, one person is sick, and the other one cares about his/her health.			
Conversation			
Speaker	Utterance	Emotion	Intent
1	你今天来得比平时晚呀。是身体不舒服吗？ (You are later than normal days. Are you OK?)	caring	ask
2	呜呜，昨晚空调开的太，一大早起来头就特别疼。 (Yesterday the air conditioner was too cold that I had a headache this morning.)	sad	description
1	怪不得，那你吃过感冒药了吗？ (I know. Have you taken the medicine?)	caring	ask
2	吃过了，现在已经好多了，就是有点想睡觉。 (Sure. I feel better now, just feel a little bit sleepy.)	other	description
...	...	...	...
2	今天的工作安排多么？ (What are today's arrangements?)	other	other
1	我会帮你做的。你好好休息吧！ (I will help finish them. You'd better take a good rest.)	caring	advise
2	真是太感谢你了！ (I really appreciate a lot for your help!)	joy	other

Table 8: Example Conversation with Annotations. Note that the underlined words stand for the emotion cause span. Words are shorten due to space limit.

Topic	Scenario
Study	两个学生之间，讨论课业压力大，总是做不完作业 (Between two students, discuss the overload homework) 考研失败，向朋友倾诉自己的伤心和烦恼 (Fail the entry exam of graduate study, express the distress to a friend)
Entertainment	讨论自己最喜欢的一部电影，以及为什么喜欢它 (Discuss one of your favorite films and why) 聊一聊自己曾经单曲循环过的歌曲，以及当时自己的感受 (Talk about a music or a song you have put on repeat all the night)
Love	情侣之间，因为生活作息不一致而吵架闹别扭 (Between a couple, quarrel with the lover due to inharmonious habits) 自己订婚了，激动地与好友分享喜讯 (Being engaged, share the good news to the best friend)

Table 9: Example Topics and Scenarios.

Intent Type	Definition	Example
ask	to know further details or clarify	*What happended?*
describe	present more details and explain the reasons	*I'm sad because I failed the exam.*
advise	give explicit solutions	*Try to exercise more.*
opinion	share own thoughts	*I don't like being disturbed after work.*
console	pacify others	*I hope you'd feel better.*
other	-	*Goodbye.*

Table 10: Annotation Criteria for Response Intent.

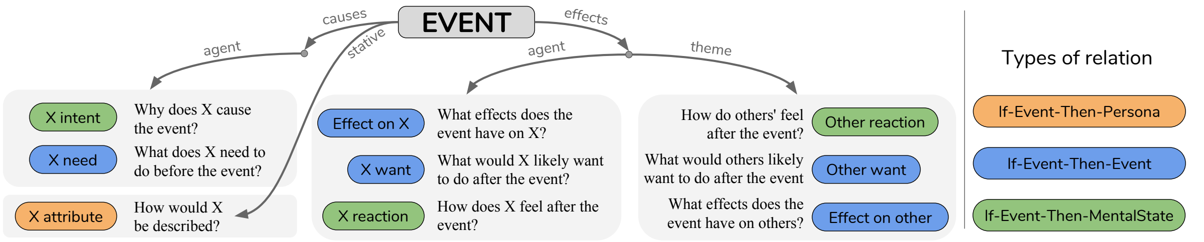

Figure 6: The taxonomy of *if-then* reasoning types. We consider nine *if-then* relations that have overlapping hierarchical structures as visualized above. One way to categorize the types is based on the type of content being predicted: (1) **If-Event-Then-Mental-State**, (2) **If-Event-Then-Event**, and (3) **If-Event-Then-Persona**. Another way is to categorize the types based on their causal relations: (1) **"causes"**, (2) **"effects"**, and (3) **"stative"**. Some of these categories can further divide depending on whether the reasoning focuses on the "agent" (X) or the "theme" (Other) of the event.

Event	Type of relations	Inference examples	Inference dim.
"PersonX pays PersonY a compliment"	If-Event-Then-Mental-State	PersonX wanted to be nice PersonX will feel good PersonY will feel flattered	xIntent xReact oReact
	If-Event-Then-Event	PersonX will want to chat with PersonY PersonY will smile PersonY will compliment PersonX back	xWant oEffect oWant
	If-Event-Then-Persona	PersonX is flattering PersonX is caring	xAttr xAttr
"PersonX makes PersonY's coffee"	If-Event-Then-Mental-State	PersonX wanted to be helpful PersonY will be appreciative PersonY will be grateful	xIntent oReact oReact
	If-Event-Then-Event	PersonX needs to put the coffee in the filter PersonX gets thanked PersonX adds cream and sugar	xNeed xEffect xWant
	If-Event-Then-Persona	PersonX is helpful PersonX is deferential	xAttr xAttr
"PersonX calls the police"	If-Event-Then-Mental-State	PersonX wants to report a crime Others feel worried	xIntent oReact
	If-Event-Then-Event	PersonX needs to dial 911 PersonX wants to explain everything to the police PersonX starts to panic Others want to dispatch some officers	xNeed xWant xEffect oWant
	If-Event-Then-Persona	PersonX is lawful PersonX is responsible	xAttr xAttr

Table 11: Examples of **If-Event-Then-X** commonsense knowledge present in Sap et al. (2019). For inference dimensions, "x" and "o" pertain to PersonX and others, respectively (e.g., "xAttr": attribute of PersonX, "oEffect": effect on others).

D.2 More Examples of Constructed Scenario Graphs

In this section, we visualize more snippets of the scenario graphs. They are "insomnia" in Figure 8,

Please kindly note that for clarity, we only visualize a small set of relation and tails in each figure, and try to give a comprehensive view of the relations by showing different relations in different scenario graphs. In fact, every scenario graphs contain the full set of C^{3}KG relations.

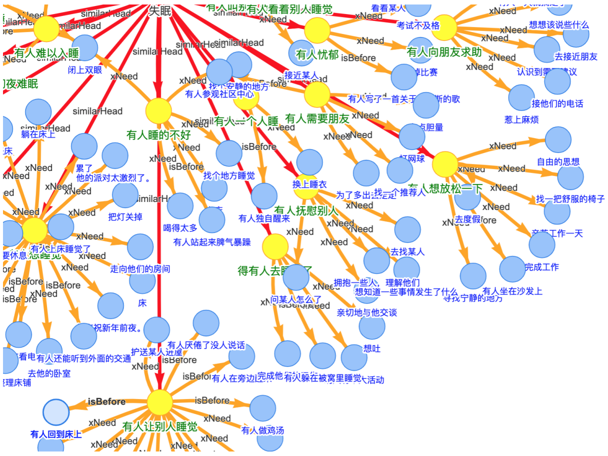

Figure 7: Scenario Graph of "Insomnia".

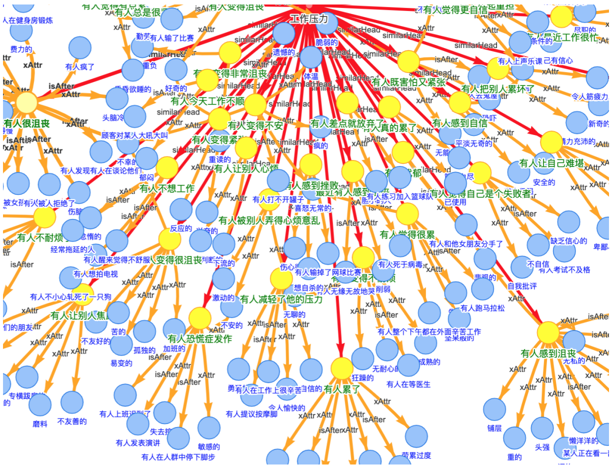

Figure 8: Scenario Graph of "Work Pressure".

Original pattern	Replaced pattern
PersonX...PersonX...	Someone...himself...
PersonX...PersonY...	Someone...some one else...
PersonX...PersonX's...	Someone...his...
PersonX...PersonY's...	Someone...someone else's
...___...	...something...

Table 13: Pattern replacement we use when translating ATOMIC

POS sequence	Example
v+v	想睡觉 (want to sleep)
v+n	做作业 (do homework)
v+i	感觉如释重负 (feel relieved)
v+u+z	跑得飞快 (run fast)
v+u+m	看了一下 (take a look)
v+c+v	讨论并通过 (discuss and approve)
v+c+i	尝试但一无所获 (try but find nothing)
a+v	热烈鼓掌 (applause warmly)

Table 12: POS templates we use in event extraction method *POS*.