CIS 700 HW5: COMET-ATOMIC Schema Report

Weiqiu You, Yifei Li

March 21, 2022

1 Story Tracking Questions

See the code.

2 Schema Explanation

Sol. We initialize the schema by the first sentence, then update it by the remaining sentences sequentially. Specifically, for each sentence, we provide a candidate set of relation preconditions and effects from COMET and consider they one by one.

For each precondition input, we use the sentence BERT to compute the embedding similarity score with each statement in the current state. If the similarity of the precondition and any statement in the current state is larger than 0.7, then we would put it into the precondition list to ensure the validity of the event. We then also add the effects of the new statement to modify the current state.

3 Answers

1. What types of stories is COMET-ATOMIC *good* at tracking? In other words, what types of information is it good at modeling? (It might help to think about how COMET-ATOMIC compares to other knowledge bases.)

   Sol. COMET-ATOMIC is generally good at tracking: (1) stories where the steps are connected closely so the gap wouldn't be too large to apply the if-else relationship in the knowledge graph (2) stories that have ordinary background and common-sense-based otherwise it would be out of the scope of the knowledge graph

2. What types of stories is COMET-ATOMIC *bad* at tracking?

   Sol. COMET-ATOMIC is generally bad at tracking: (1) stories that jump too fast or lack logic. (2) stories that have a very complicated narrative or alien background. Also, since the given sentence parser is not good at extracting the pronoun as 'PersonX' or 'PersonY' so the stories which are full of pronoun-only sentences are not suitable at this system. For example, in the 3rd story, the subject switches from Amy to the girl and the girls, and the parser cannot extract the girls as the subject and referring to Amy and the girl.

   (a) Do you think any of the other knowledge bases mentioned in class could better model these?
   Sol. GPT-3 can work as complementary module to address these issues, due to its causal generation ability. In detail, we can fine-tune GPT-3 to generate a more consistent story first given the original sentences of story, then feed to COMET for the schema.

Appendix - Stories

1. **Story**: "Gina misplaced her phone.", "Gina looks for her phone in the living room.", "Gina remembers leaving her phone in the car.", "Gina goes back to the car.", "Gina finds her phone in the car."
   Schema: {'Gina': {'Gina misplaced her phone.', 'Gina gets out of the car', 'Gina is looking for her phone', 'tired', 'Gina finds the phone', 'to use the phone', 'to look for the phone', 'Gina looks for their phone in the car', 'Gina looks for her phone in the car', 'relieved', 'none', 'regretful', 'Gina finds her phone', 'to get in the car', 'Gina looks for the phone in the car', 'Gina puts the keys in the ignition', 'Gina looks for their phone', 'Gina puts the phone in the car', 'Gina finds the phone in the kitchen', 'Gina loses their phone', 'to find the phone', 'Gina calls the police'}})

2. **Story**: "Phil was at the community pool.","Phil thought he could go out to the deeper end by himself.","Phil jumps into the deep end.","Phil has trouble staying afloat.","The lifeguard had to help Phil out of the water."
   Schema: {'Phil': {'Phil was at the community pool.', 'to get out of the water', 'jumps into the deep end', 'to make sure they are okay', 'happy', 'Phil jumps into the deep end', 'to go to shore', 'Phil gets out of the water', 'Phil jumps into the pool', 'relieved', 'Phil drowns', 'none', 'to have fun', 'Phil goes to the swimming pool', 'Phil swims in the ocean', 'Phil jumps in the water', 'the lifeguard helps Phil', 'Phil swims in the pool', 'Phil falls into the water', 'to swim in the pool', 'sad', 'the lifeguard gets wet'}}

3. **Story**: "Amy was happy her first class in junior high was all new kids.", "Amy introduced herself to the girl seated next to her.", "The girl was even more nervous than Amy to make friends.", "The girls talked and bonded over their love of books.", "The girls decided to meet up after school to go to the library."
   Schema: {'Amy': {'Amy makes new friends', 'none', 'gets to know the girl', 'to get to know them', 'Amy meets a new girl at school', 'Amy makes friends with the girl', 'Amy asks the girl out on a date', 'to talk to the girl', 'Amy smiles at the girl', 'Amy meets a new girl', 'Amy was happy her first class in junior high was all new kids.', 'nervous', 'Amy sees a girl sitting next to her', 'friendly', 'to talk to Amy', 'Amy gets nervous'}}
   (Note: this story skipped the last two sentences due to the failure of Sentence Parser on pronoun extraction.)

4. **Story**: "Xander's dog hates his treats.", "Xander decided to go buy some new dog treats.", "None of the dog treats at the pet store looked tasty.", "Xander decided to buy his dog some salmon from the fish market.", "Xander's dog loved the salmon."
   Schema: {'Xander': {"Xander's dog hates his treats.", 'to feed the dog', 'to eat the salmon', 'Xander takes the dog to the market', 'none', 'to feed it to the dog', 'Xander feeds the dog salmon', 'Xander takes the dog to the fish market', 'Xander gives the salmon to the dog', 'Xander feeds the dog', 'Xander feeds the salmon to the dog', 'happy', 'Xander buys dog treats', "Xander's dog ate all the treats", 'Xander gives the treats to the dog', 'Xander has a dog that is sick', "Xander's dog eats salmon"}}
   (Note: this story skipped the third sentence due to the failure of Sentence Parser on pronoun extraction.)

5. **Story**: "Tim has never cooked for his family.", "Tim decided to follow an old family recipe.", "Tim's grandma told him anybody could make the recipe.", "Tim made a whole meal for his family in one hour.", "Tim's family all loved the meal."
   Schema: {'Tim': {'to feed the family', "Tim's grandma makes the recipe", 'Tim cooks a meal', 'Tim is thanked', "Tim's grandma taught him how to cook", 'to share the meal', 'eats the meal', 'Tim wants to try a new recipe', 'Tim makes dinner for his family', 'they eat the meal', 'Tim eats the recipe', 'satisfied', 'Tim has never cooked for his family.', 'Tim wants to learn how to cook', 'to eat the recipe', 'none', 'Tim serves the meal to their family', 'to thank Tim', 'Tim wants to make a new recipe', 'Tim eats dinner with the family', 'Tim cooked a delicious meal', 'happy', 'proud', 'Tim feeds the family dinner'}}