Is Concept Flow useful concept enough for open dialogue domain?

Anonymous Authors[1]

Abstract

Most dialogue generation models used memory networks to remember previous subjects of the conversation. However, most memory networks contain the inputs linearly in memory while encoding and decode it to create dialogues. This means that memory networks create dialogues from past conversations and give responses in context with the given topic, but cannot generate topics other than what have been discussed. Therefore, we explored Concept-Flow as a solution to recover the limitations of the memory network. Conceptflow can generate conversation on new topics by using a knowledge graph, which embeds diverse information according to relationship among the words. In order to understand Conceptflow and how it works, we decided to study the specific example of the Dialogue generation model. We have replicated the model with most of its basic functions and by doing so, we could have a thorough understanding of how Conceptflow works. We also created graphs that are generated when implementing dialogue generations based on Conceptflow. The graph generated showed how each word for dialogues were chosen. We also found that the dialogues generated was not smooth and they had similar structures starting with common words. In order to resolve this, we need better computing resources such as GPU, and larger datasets.

1. Introduction

Have you ever wondered about robots who can chit chat with humans like a friend? Open-domain dialogue generation aims to generate dialogues that can satisfy the human need for communication, affection, and social belonging.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

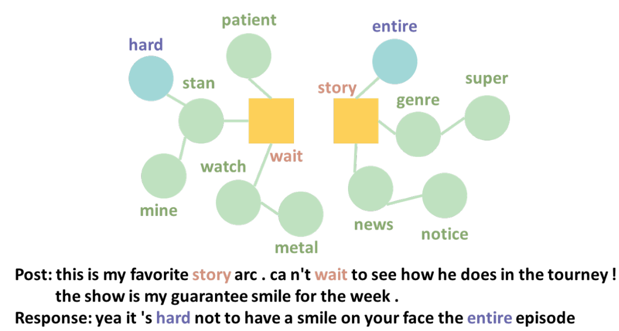

Figure 1. Part of ConceptNet graph. It shows how the words are chosen.

Indeed, the open dialogue creation model has been continuously evolving. Google announced Meena in January 2020 and Facebook announced the Blender bot in July 2021. These models enable more sophisticated, human-like conversations.

Before Meena and the Blender bots existed, many researchers have studied diverse chat-bot models for open-domain dialogue models. To create a human-like model, it is important to catch the subject of the conversation and transfer it to related subjects naturally. Therefore, many models use a memory network to remember previous subjects of the conversation.

Memory networks contain the inputs linearly in memory while encoding them and use the information during decoding to create output dialogues(4). Memory networks can effectively extract keywords from past conversations, but cannot generate topics beyond what has already been discussed. Also, the linear architecture of the memory network does not adequately capture the relationships between subjects. Conversations often develop around Knowledge. A promising way to address the degeneration problem is to ground conversations with external knowledge such as open-domain knowledge graph, commonsense knowledge base, or background documents. Recent research leverages such external knowledge by using them to ground conversations, integrating them as additional representations, and then generating responses conditioned on both the texts and the grounded semantics.

Integrating external knowledge as extra semantic representations and additional inputs to the conversation model effectively improves the quality of generated responses.

In this mini-project, we explored ConceptFlow(1) as the solution to recover the limitations of the Memory network. The objective of this model is to construct an algorithm that can effectively reflect the relationship between topics and easily cross over various topics. It uses a knowledge graph, which embeds diverse information according to the relationship among the words, to generate conversation about various subjects. By using graph structures as embedding topics, we can easily grasp the connection between topics and search for new content easily. We focused on the following three key points in the process of replicating the ConceptFlow model. The first objective is to study the specific example of the Dialogue generation model associated with Open Dialog by replicating it. The second objective is to create graphs that are generated while implementing dialogue generation based on the Conceptflow model. The last objective is to understand the limitations of the model and find the resolutions.

ConceptFlow leverages commonsense knowledge graphs to model the conversation flow in the explicit concept space. For example, as shown in Figure 1, the given topic story and wait is connected in the graph with other related topics, such as patient or news, and these keywords can be incorporated in as the next response. To better capture this conversation structure, ConceptFlow explicitly models the conversations as traverses in commonsense knowledge graphs: it starts from the grounded concepts and generates more meaningful conversations by hopping along the commonsense relations to related concepts.

The traverses in the concept graph are guided by graph attention mechanisms, which derives from graph neural networks to attend on more appropriate concepts. ConceptFlow learns to model the conversation development along more meaningful relations in the commonsense knowledge graph. As a result, the model is able to “grow” the grounded concepts by hopping from the conversation utterances, along the commonsense relations, to distant but meaningful concepts; this guides the model to generate more informative and on-topic responses. Modeling commonsense knowledge as concept flows, is both a good practice on improving response diversity by scattering current conversation focuses to other concepts, and an implementation solution of the attentional state mentioned above.

2. Related Work

For the development of Natural language processing (NLP), researchers got their idea from how humans communicate. For example, COPYNET applied a human conversational pattern to the SeqtoSeq model to re-state expressions used in previous conversations(9). It introduced integrated algorithms for copying the chosen sub-sequence of input data and using it for decoding.

Efforts to communicate as humans lead to expanding the research area to building a model that people can talk with about open domain. In general conversation, humans tend to cross over various topics based on their knowledge. Efforts to communicate as humans lead to expanding the research area to building a model that people can talk with about open domain. In general conversation, humans tend to cross over various topics based on their knowledge. Therefore, the Generative dialogue system(GenDS) introduces the conversation generation based on the Knowledge-Based(KB) to eliminate the limitation that the previous study cannot deal with out of vocabulary entities. Gen is the fully data-driven generation method by searching the KB related to the input. However, it didn’t show the relationship of entities that make up the knowledge graph, because it was approached separately rather than from a perspective of the entire graph. Therefore Commonsense Knowledge aware conversational model(CCM) launched two novel graph attention algorithms to use the relations of the entities: a static graph for understanding the hidden meaning of a post and a dynamic graph for generating the semantic response(3).

The studies to use and develop graph concepts have continued for creating natural conversations. Graphs of Relations between Facts and Text Networks (GRAFT-Net) presented a convolution-based model for generating links between KB facts and linked texts for the open domain Question-Answer(**?**). OpenDialKG Walker model has the mechanism to learn the paths in KB with the paralleled dialog(4). Unlike previous studies, We tried to replicate the model to use multi-hop concepts.

3. Solution

ConceptFlow first constructs a concept graph G with central graph $G_{central}$ and outer graph G_{outer} according to the distance (hops) from the grounded concepts. Then ConceptFlow encodes both central and outer concept flows in central graph $G_{central}$ and outer graph G_{outer} , using graph neural networks and concept embedding. The decoder leverages the encodings of concept flows and the utterance to generate words or concepts for responses.

3.1. Concept Graph Construction

ConceptFlow constructs a concept graph G as the knowledge for each conversation. It starts from the grounded concepts (zero-hop concepts V_0), which appear in the conversation utterance and annotated by entity linking systems. Then, ConceptFlow grows zero-hop concepts V_0 with one-hop

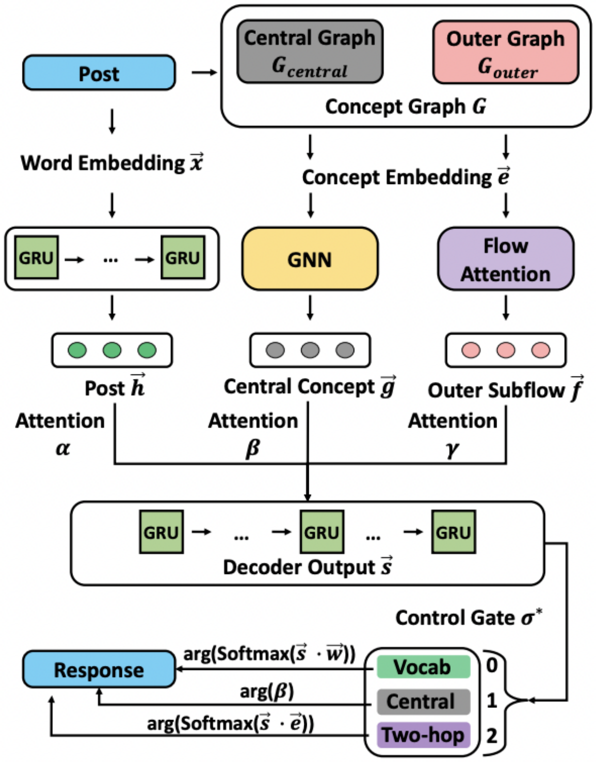

Figure 2. Overview of the algorithms in ConceptFlow.(1)

concepts V_1 and two-hop concepts V_2. Concepts from V_0 and V_1, as well as all relations between them, form the central concept graph $G_{central}$, which is closely related to the current conversation topic. Concepts in V_1 and V_2 and their connections form the outer graph G_{outer}.

3.2. Encoding Concept through Concept Graph

The constructed concept graph provides explicit semantics on how concepts related to commonsense knowledge. ConceptFlow utilizes it to model the conversation and guide the response generation. It starts from the user utterance, traversing through central graph $G_{central}$, to outer graph G_{outer}. This is modeled by encoding the central and outer concept flows according to the user utterance.

Central Flow encoding. The central concept graph $G_{central}$ is encoded by a graph neural network that propagates information from user utterance H to the central concept graph. Specifically, it encodes concept $e_1 \in G_{central}$ to representation. There is no restriction of which GNN model to use. We choose GraftNet for GNN, which shows strong effectiveness in encoding knowledge graphs.

Central Flow encoding. The outer flow f_{e_p}, hopping from $e_p \in V_1$ to its connected two-hop concept e_k, is encoded to $\vec{f_{e_p}}$ by an attention mechanism:

$$\vec{f_{e_p}} = \sum_{e_k} \theta^{e_k} \cdot [\vec{e_p} \frown \vec{e_k}]$$

where $\vec{e_p}$ and $\vec{e_k}$ are embeddings for e_p and e_k, and are concatenated ($\frown$). The attention θ^{e_k} aggregates the concept triple (e_p, r, e_k) to get $\vec{f_{e_p}}$:

$$\theta^{e_k} = softmax\left((\omega_r \cdot \vec{r}) \cdot \tanh\left(\omega_h \cdot \vec{p} + \omega_t \cdot \vec{k}\right)\right)$$

where $\vec{r}$ is the relation embedding between the concept e_p and its neighbor concept e_k . ω_r, ω_h and ω_t are trainable parameters. It provides an efficient attention specifically focusing on the relations for multi-hop concepts.

3.3. Generating final output

To consider both user utterance and related information, the texts from the user utterance and the latent concept flows are incorporated by decoder using two components: 1) the context representation that combines their encodings; 2) the conditioned generation of words and concepts from the context representations.

model	total ppl	word ppl	one-hop ppl	two-hop ppl
3	91.006	102.865	38.381	35.526
4	89.1145	100.6442	37.888	34.826
5	88.965	100.71	37.229	34.493
6	89.734	101.459	37.701	34.401
7	91.928	104.122	37.994	34.6537
8	95.986	108.579	40.3517	35.27

Table 1. These are the perplexiity score of response generated.

4. Experiment Methods

4.1. Dataset

The dataset used is extended version of single-round dialogs. It contains multi-hop conversation having 3,384,185 train data and 10,000 test data. Since the data is too big to handle with provided resource, we used 100,000 data for training and 10,000 data for testing. It contains the knowledge graph information made from ConceptNet. It has 120,850 triples which are made of 21,471 nodes and 44 relation edges. For exmaple, the nodes *table* and *setting* are connected with the edge *RelatedTo*. Figure 1 shows how the response is created based on the post. In the graph, it chooses appropriate words in one-hop and two-hop.

4.2. Evaluation Methods

For the evaluation of the project, we used both of human evalution and Concept-PPL to compare Relevance. Concept-PPL means combining the distribution of both words and

Post	this fucking shit pisses me off to no end , when these fucking liberal hypocrites imply the only group of people capable of being racist are the whites . hh can be used to put pressure on your opponent.
Response	i do n't think you 're sexist . i do n't think that 's a lot of racism .
Post	so not just pissing off stronger rivals , if your hh is strong enough you can make them weak in the battle.
Response	i 'm not sure if you 're a good rival .
Post	today i stopped to make a left turn . the jerk behind me was honking at me for waiting for a group of kids to cross to the community centre . people are assholes .
Response	i do n't think you 're going to be a asshole . you do n't want to be a asshole . you 're a good game .
Post	i wear glasses and at the age of 17 i would have been mortified to be seen in a photo wearing them .
Response	i do n't know if you 're a glass . i do n't think you 're the only one in the
Post	will you be requiring a shake today , sir ?
Response	i 'm not a sir .

Table 2. These are the three examples of response generated.

model	dialogue
3	i 'm not sure that 's a hypocrite . i 'm not sure that 's a rival . i 'm not sure that i 'm going to play the game . i 'm not sure that i 'm going to wear the glasses .
5	i do n't think you 're the only thing to do that . i do n't think the hypocrite is n't the same i 'm not sure if you 're a rival , but i do n't think you 're a good job . i do n't know if you 're going to be a asshole . you 're a good thing . i do n't know if you 're going to be a good thing . i do n't know if you 're a good guy . i do n't know if you 're a good guy
8	i do n't think you 're capable of the racist . i 'm not a hypocrite . i 'm a hypocrite . i 'm not sure if you 're a rival , not the same . i do n't know if you 're in the same way . you 're a good game , you 're a good game . i do n't know if you 're a glass . i do n't think you 're the only one in the

Table 3. These are the three examples of response generated.

concepts together. Human evaluation is also precious because this project aims at building a model which can communicate like a human. Therefore, we used informativeness and appropriateness like the metrics of original project, but as the slightly modified way.

Concept PPL Concept-PPL is the method used in Commonsense Knowledge aware conversational model. it used the both words perplexity and entity score. Words perplexity Evaluates the model at the content level, which is whether the content is grammatical and relevant in topic. Also, entity score calculates the number of entities per response to measure the model's ability to select the concepts from the commonsense knowledge base in dialogue generation procedure.

Human evaluation method Human evaluation is precious because this project aims at building a model which can communicate like a human. Therefore, we used informativeness and appropriateness like the metrics of original project, but as the slightly modified way. The 6 sample responses to the same post are randomly selected from each epochs for evaluation. A total of 10 students participated in this test. Evaluators are required to give the response sentences score from 1 to five based on the following three metrics: Informativeness, appropriateness of grammar, and topic.(3).

5. Result

5.1. Perplexity Score

Table 1 is the perplexity score of the generated result. As mentioned in the evaluation method, perplexity score is calculated through word perplexity and entity score. One-hop ppl is the perplexity score for using only one-hop concepts, and Two-hop ppl is the perplexity score for using only two-hop concept. Model number shows the number of train iterations run. In Table 1, we can see that model 5 has the highest evaluation result, and that as the number of iteration increases, scores are increasing as well.

5.2. Qualitative analysis of Generated Dialogue

Table 2 is generated dialogue cases. Three cases are not directly answering to the question, and the sentences don't make sense. Case 1 and 2 starts with *i'm not*, and case 3 is not in a correct sentence. Analyzing with 1,000 test data, the 716 responses of the generated dialogue started with *i'm*. Among them, 428 sentenses contained *i'm not sure*, and 41 had *i don't know*. The responses generated were ambiguous and they had simple structure.

In Table 2, The highlighted parts show the characteristic of generated output. The blue highlight in **response** shows the words used in **post** and was used again in the output. The red highlight shows the words that was not included in the input, but was related with the concepts in the input. The teal highlight shows repeating parts of the generated dialogue, which is possibly because the model is less trained.

The positive side of the generated dialogue is that it Uses similar words in input such as Racist, asshole, sir, glasses, and tries to use concepts relates with input that is not already given. However, there are lots of limitations such as that same phrases appear repeatedly and words are repeatedly used within one sentence, and there are lots of grammar errors. Such limitations may be solvable by increasing the epoch number or tuning the model with more appropriate parameters.

5.3. Human evaluation

Appropriateness of Grammar Participants checks whether the response is appropriate only based on the grammatical accuracy. In table 4, there are the average points of the responses in each epoch. Most of the results couldn't get good results because the rules of capitalization and space were wrong. Epoch 1 and 8 got the lowest score because epoch 1 includes same vocabulary redundantly and epoch 8 include 2 unfinished sentences.

Appropriateness of topic Table 4, there is the result of whether the model generates responses related to the topic. Evaluators commonly comment that most of the sentence is contextually weird. However, epoch 3 is among the results that get the best average score. Participants replied that the answers generated were expressions that can be easily used in various contexts such as "I am not sure."

Informativeness For this metric, the questionnaire asked to give a score based on the judgment of the evaluator thinks the response includes any new information. The result is quite interesting because the more model is trained, the higher score is. In the interviews, respondents state that they observed longer sentences and more new vocabularies.

Epoch	Grammar	Topic	Info.
1	2	2	1
2	2.5	2	2
3	2.5	3	2
4	2.5	1.5	1
5	2.5	1.5	2.5
6	2.5	1	3.5
7	2.5	0.5	3.5
8	2	0.5	4

Table 4. This is the average score about appropriateness of grammar(Grammar), appropriateness of topic(Topic) and Informative(Info.)

5.4. Comparison between differently trained models

Table 3 shows the different responses generated by models trained in different iterations. As shown in Table 2, model 8 has the highest total perplexity loss, and model 5 has the lowest perplexity loss. Comparing model 3, 5 and 8, it is clearly shown that the variations of word choices and topics are improving as the the model is more trained. Also, precision in grammar is more accurate in model 5 and 8 compared to model 3.

6. Conclusion

We used ConceptFlow for dialogue generation. Through the ConceptNet, it generated edges between related nodes. The dialogue generated by our model was not smooth and they had similar structures repeating the same words. The training data was small, so it may be hard to learn to select the word and generate response sentences. This problem occurred by the limitations of computing resources since ConceptNet has a big size of data. With bigger GPU memory, more data can be used for training, and it will lead to higher performance. Although we could not use full data, we could analyze how the model generates the sentences and chooses new topics through this project.

References

[1] Zhang, Houyu, et al. "Grounded conversation generation as guided traverses in commonsense knowledge graphs." arXiv preprint arXiv:1911.02707(2019).

[2] Sun, Haitian, et al. "Open domain question answering using early fusion of knowledge bases and text." arXiv preprint arXiv:1809.00782 (2018).

[3] Zhou, Hao, et al. "Commonsense knowledge aware

conversation generation with graph attention." IJCAI. 2018

[4] Moon, Seungwhan, et al. "Opendialkg: Explainable conversational reasoning with attention-based walks over knowledge graphs." Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019.

[5] Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In Proceedings of ACL, pages 1631–1640.

[6] He, He, et al. "Learning symmetric collaborative dialogue agents with dynamic knowledge graph embeddings." arXiv preprint arXiv:1704.07130 (2017).

[7] Tuan, Yi-Lin, Yun-Nung Chen, and Hung-yi Lee. "Dykgchat: Benchmarking dialogue generation grounding on dynamic knowledge graphs." arXiv preprint arXiv:1910.00610 (2019).

[8] Liu, Zhibin, et al. "Knowledge aware conversation generation with explainable reasoning over augmented graphs." arXiv preprint arXiv:1903.10245 (2019).

[9] Gu, Jiatao, et al. "Incorporating copying mechanism in sequence-to-sequence learning." arXiv preprint arXiv:1603.06393 (2016).