Neural Language Model for Argumentative Response Generation

[redacted]

Abstract

In this project, I present Argue Bot, a system of generative neural language models for argumentative response generation. The aim of this project is to generate quality responses to a given input statement with relevance to the topic, coherence in the language, clear and consistent stance and logic and variability. The model is trained on quality argumentative conversations sourced from Reddit to outperform a basic baseline model.

1 Introduction

The goal of this project is to generate argumentative responses to user inputs. The stance of the generation should be controllable, meaning that the user should be able to choose whether the model should generate a counter argument or a supportive argument. Loosely, the task can be defined as: given user input $x = x_0, x_1, ..., x_n$ and f, where x is an argumentative statement in a sequence of English word tokens and f is a binary label in $\{supportive, contradicting\}$, the model will produce y, a response in one or more English sentences which argue around the topic of x with a clear stance indicated by i. Moreover, the input should be simple, clear and arguable, not a widely-accepted-or-rejected factual statement(e.g., *"the moon rotates around the earth every 29 days"*) or a subjective statement(e.g., *"I like eating hamburgers"*). A valid example is *"Bar/nightclub culture would be more fun/safer for women if there were more gay men harassing straight men"*

Ideally, the generated text should reflect coherence in the language and relevance to the topic. In addition, the argument should be logical and hold a clear and consistent stance chosen by the user.

Common approaches to takle this task can be classified into 1. Rule based generation 2. Information retrieval 3. Generative neural language model. Rule-based argumentation planning had been explored as early as in 1996. (Reed et al., 1996) In a rule based system, one typically breaks down the input argument into claims, reasons and evidence. Then they can search in a knowledge base for information that supports or contradicts the claim. With this information, the system then formulate a response following an argumentative pattern. This type of generation usually guarantees the coherence of the sentence and has a strong information. However, the text generated usually lacks of variance. Even the style of "argument" is the same. The second approach detects the topic and the stance of the input argument, then search in an argument database for related arguments. It usually applies minimal or no modification to the retrieved argument. This approach also lacks of variability and, more importantly, is not easily scalable and can't be easily generalized to generate arguments on any topics because the argument database does not scale well. A good representative of this type of system is End-to-end Argument Generation System in Debating (Sato et al., 2015).

2 Data

The primary dataset used is the Winning Arguments (ChangeMyView) Corpus (Tan et al., 2016) from Cornell Convokit 2.4. This dataset includes 3051 conversation threads containing 293297 individual utterances posted by 34911 unique users on the r/changemyview subreddit from January 1st 2013 to May 7th 2015. Later on, I also scrapped more data from r/changemyview on my own to enrich the topic coverage in the dataset.

2.1 r/changemyview

r/changemyview is a subreddit where people post their opinion about a certain topic, and look for other's different views on the subject in the responses to their post. The discussion on r/changemyview ranges from controversial topics such as politics and sexuality to ordinary day-to-day matters. What makes r/changemyview an ideal data source is the ”delta system”(Δ). By the rule of r/changemyview, if a response to a post is deemed so convincing such that it changes one’s mind about the topic, one should award that response a ”Δ” by replying ”Δ” to the response. Therefore, the high-quality arguments are naturally marked by human evaluators.

Another thing to point out about r/changemyview is that the users who want to make a post are required to put a clear, brief statement summarizing their view in the title prepended with the tag ”CMV:”. This rule guarantees that every post in this subreddit with a prefix ”CMV:” in the title is verified by a moderator to be a valid, arguable statement, conforming with my assumption made in section 1 about the validness of the user input.

2.2 Pre-processing

The Winning Argument dataset is highly structured and well labeled. To leverage pre-trained generative neural language models, I cast this task to conditional next sentence prediction: given a claim, the model generates the next sentence(s) which is follow-up argumentation around the claim, depending on the binary condition i, stance. Since each instance of a conversation in the dataset has a central claim (the title of the post), follow up arguments (body text of the post) and counter arguments (responses from other users), we can put the utterances into claim-argumentation pairs by matching the post title with the original post body and the responses. I also applied a data filtering strategy similar to it in DialoGPT (Zhang et al., 2020). This includes removing samples that has embedded URLs, mark-up languages, highly repetitive meaningless content, offensive content and so on. All post bodies and responses are truncated to a maximum of 4 sentences ended with a period ”.”, a question mark ”?” or an exclamation mark ”!”.

2.3 Statement Negation Generation

There are arguments taking either stance with the original claim. To make the model learn to take a consistent stance, we need to maintain the stance consistency of the input claim-argumentation pairs. To address this issue, I introduced a trick to augment the data, that is for each conversation thread, generate the negation of the original title. Since the original poster (OP) would always argue for the claim in the title and other users would always argue against it in the responses, we can pair up the title with the original post body, and the negation of the title with the responses. This way we can get claim-argumentation pairs where the argumentation always argues for the claim and the model can be trained to always generate supportive arguments to an input statement.

During generation, we can control the stance of the generated argument by selectively taking the negation of the user input sentence depending on the chosen stance.

To generate the negation of a sentence, I made a naive rule-based python script. It first looks for any linking verbs (e.g. is, am, been, seem, ...), auxiliary verbs (e.g. should, must, can, ...) and ”do”’s (i.e. do, did, done) as well as their negation forms (e.g. isn’t, shouldn’t, don’t, ...) Then it removes or injects a negation, usually the word ”not”, to the sentence. It also uses nltk part-of-speech tags to ensure the tense of the verb stays the same. Admittedly, such a simple rule-based sentence negator is unable to handle any sophisticated sentence structures. Therefore when the sentence negator sees any sentence with multiple auxiliary verbs or linking verbs, connecting clauses such as ”if”, ”because”, ”therefore” and ”, and”, the negator skips the sample.

Lastly, to further ensure the correctness, a grammar checker checks the output of the sentence negator. If the grammar is not correct, I also discard the sample from the pipeline. Overall, the grammar checker reports a 94.3% correctness of the negator. Nevertheless, false negatives may exist because the grammar checker only checks for the superficial grammar correctness while the negator might have changed the meaning of the sentence when injecting/removing a negation word. Sometimes it is even difficult for humans to decide what the negation form of a statement should be (e.g. ”!”), and the negator script might fall in these cases too.

2.4 Reddit Scraping

With the data pre-processing pipeline set up, I further collected more data from r/changemyview by using PRAW reddit scraping API. The data was collected in a format similar to the Winning Argument Corpus with conversation threads. More up-to-date data not only enrich the topic coverage of the model but also exposes the model to recent discussions items such as the Biden presidency in 2020.

3 Generation

3.1 Baseline

Though there exists more sophisticated argumentation generation system, I have chosen to use a generic chat-bot style generative language model as the baseline for a few reasons: 1. it is highly accessible to use and evaluate. 2. it fulfills the premise of using pure neural model 3. in a natural conversation, one can expect to get the opinion of the other participant of the conversation about a specific matter by prompting with activating clauses such as ”..., what do you think?”, ”..., what’s your opinion?”. This idea goes along with the goal of a general purpose chat bot. 4. chat-bot style generative language model allows for fine-tuning with custom dataset to adapt the model to a specific task.

The model I have chosen is DialoGPT (Zhang et al., 2020), which has an architecture inherited from GPT-2 (Radford et al., 2019). The baseline model is then re-trained with conversations from the Winning Arguments dataset: each conversation thread in the dataset is modeled as a conversation tree where the root node is the title of the post plus the first four sentences in the main post body. Then each response and nested response is a child node in the tree. Each unique walk in the conversation tree from the root to a leaf forms a conversation thread and is used to fine-tune the DialoGPT model.

3.2 Argue Bot

The base model of Argue Bot inherits from the Open AI GPT-2 (Radford et al., 2019) model with 24 layers (mid-size) and 345M parameters. It is pre-trained with 40GB of high-quality Reddit articles. Since I am focusing on a single-turn response generation, the training task is next sentence prediction: given a claim-argumentation pair $C = c_0, c_1, ..., c_N$ and $A = a_0, a_1, ..., a_M$, where C is a claim in a sequence of English word tokens and A is a follow-up argumentation. Denoting the concatenation of C and A with Y, the model learns to maximize the probability of (1)

$$p(Y|C) = \prod_{i=N+1}^{M+N+2} p(y_i|y_0, ..., y_{i-1}) \qquad (1)$$

As mentioned in previous sections, the condition i can be omitted here because we can simply negate the input statement C to ask for a counter argument. I also introduced a new special token ”⟨| *sep*| ⟩” to the model to prompt the model to start generating argumentation by inserting it after every claim sequence.

3.3 Negation Token

As confirmed in human evaluation, the effectiveness of the negation generation is limited. The model lacks the ability to reason and hold a consistent stance around an argument. Further more, the negation generation trick is inelegant on its own as it is often incapable of negating a sophisticated sentence correctly, resulting in the loss of training data and limits to the complexity of the user input statement. In CTRL (Keskar et al., 2019), the authors models the text generation task such that it is conditioned on not only the context sequence X, but also a separate conditional variable c provided at the input of the model. Inspired by CTRL, I adjusted the training procedure by replacing the separator token ”⟨| *sep*| ⟩” with two new tokens ”⟨| *sup*| ⟩” and ”⟨| *con*| ⟩”, meaning supportive argument and counterargument. Because the separator token did not exist in the pre-training stage of the GPT-2 model, this change would not hurt the base model’s performance.

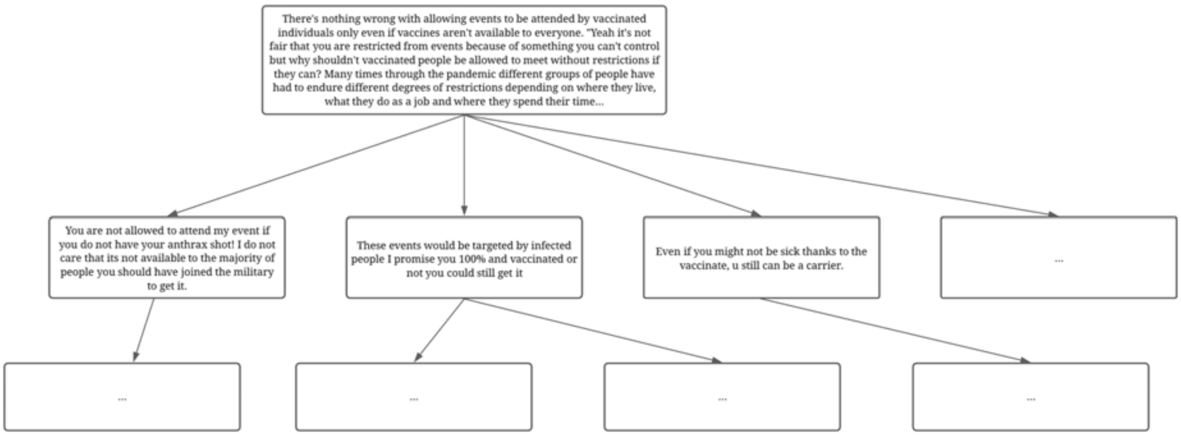
With the additional conditional token, the task is now modeled by (2), where f is a binary label in $\{supportive, contradicting\}$

$$p(Y|C, f) = \prod_{i=N+1}^{M+N+2} p(y_i|y_0, ..., y_{i-1}, f) \qquad (2)$$

After retraining the model, I performed human evaluation to a pre-selected samples and saw no significant improvement. This would be supported by the filtering model elaborated in the next chapter. My explanation is that the distribution of two types of arguments completely overlaps with each other. Intrinsically an argumentation cannot be defined as ”counterargument” or ”supportive argument” without a context. In other words, the same argument may be considered as a supportive argument or a counterargument depending on the context(claim). Therefore even with this conditional control code, it still falls back to requiring the model to detect the negation in the claim.

4 Stance Filtering

To further tackle the problem of inconsistent stance in the generated text, I configured the base genera-



There's nothing wrong with allowing events to be attended by vaccinated individuals only even if vaccines aren't available to everyone. "Yeah it's not fair that you are restricted from events because of something you can't control but why shouldn't vaccinated people be allowed to meet without restrictions if they can? Many times through the pandemic different groups of people have had to endure different degrees of restrictions depending on where they live, what they do as a job and where they spend their time...

You are not allowed to attend my event if you do not have your anthrax shot! I do not care that its not available to the majority of people you should have joined the military to get it.

These events would be targeted by infected people I promise you 100% and vaccinated or not you could still get it

Even if you might not be sick thanks to the vaccinate, u still can be a carrier.

Figure 1: Conversation Tree

tive model to generate multiple samples by specifying the beam search behavior and keep 10 most probable outputs, then applying a secondary regressor model to pick a best response from these 10 outputs. This way, the generative model can focus on generating relevant and coherent text, while the secondary regressor filter model improve the stance consistency.

The regressor model I used is a pre-trained RoBERTa (Liu et al., 2019) re-trained with claim-argumentation pairs on the task of sequence classification - whether the argument is supportive or contradictory to the claim in the input. The positive samples are simply claim-argumentation pairs used in the previous steps, while there are different types of negative samples. First, I re-matched the pairs such that the argument is opposing the claim in the pair, i.e. title-response and negated title-original post body. Second, I matched all unique titles with random responses/original post body excluding the corresponding one as negative samples. This is to prevent the regressor model from converging too fast and always picking neutral responses from the generator. To avoid having a overly imbalanced dataset between positive samples and negative samples, only 9937 such samples are selected.

Overall the classifier achieved an accuracy of 62.4% of correctly predicting the relationship between the claim and the argument on the test set before exhaustive parameter sweeping. Stance detection is on its own a challenging task, and has been studied by many. While I only applied basic pre-trained language model because it is the most accessible approach, there are many alternatives which can be applied in this project. In a recent study(Kobbe et al., 2020), the authors applied lexicon analysis and reported a similar-to-BERT performance on predicting the relationship between two statements. There is space for improvements in this aspect of the project, and future improvements on the accuracy of stance identification would largely improve the end result of this project.

After training, the softmax layer on the RoBERTa model is removed so that it outputs two raw probabilities of the input sample being in either class. I defined the stance score to be p(correct label) - p(incorrect label). A generation with a high difference in the probability to be in the two classes, as predicted by RoBERTa, is considered to have a clear stance.

5 Evaluation

Four major metrics which I proposed as a human evaluator are relevance to the topic, coherence in the language, clear and consistent stance and logic and variation in the generation. Relevance, coherence and variability can be tested by existing metrics, while it is relatively difficult to gauge the logicality of the argument automatically. The stance identifier I developed works as an automated evaluator for stance consistency, however the accuracy of the stance identifier model itself is limited. Moreover, the generation of ArgueBot is selected based on the filtering model, meaning the end output will definitely have a strong rating from the filtering model in order to be selected. Therefore I decided to deprecate this evaluation method. Overall, ArgueBot can generate arguments at a greater length (Table 2) and achieved a better BLEU score than the baseline (Table 3), although I do not think

BLEU score is a good metric for this task.

5.1 Human Evaluation

I kept a list of 100 sample claims for human evaluation and for each iteration of the model as well as for the baseline model, I went through these samples to gauge the performance of my model.

Although the baseline model is also fine-tuned with the same dataset, it display a style of generation close to the pre-trained model: the output is shorter and constantly in a conversation-like fashion. To account for this, I prepared two versions of input statement for the baseline model. The first one is the raw claim sentence same as the input for ArgueBot, and in the second style prompts are added to mimic a ”conversation environment”. Different prompts are used in parallel on the input, and the one resulted in the best output is chosen for evaluation. (Table 1) Over the 100 test samples, ArgueBot answered 22 of them relatively well with consistent and correct stance and at least one novel argument item (evidence or logic), while the baseline constantly generates irrelevant and unargumentative responses, often repeating the whole sentence or part of the claim. ArgueBot also answered 14 claims with informative content but wrong stance. The responses to all other samples generally lack of novel information and sometimes have logic errors.

6 Conclusion

In this project, I build a argument generation system exclusively relying on neural language models which performed better than the basic baseline model according to my judgement. Though the result is far from impressive, it has been a great learning process for me.

References

Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation.

Jonathan Kobbe, Ioana Hulpus, and Heiner Stuckenschmidt. 2020. Unsupervised stance detection for arguments from consequences. pages 50–60.

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.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Chris Reed, Derek Long, and Maria Fox. 1996. An architecture for argumentative dialogue planning.

Misa Sato, Kohsuke Yanai, Toshinori Miyoshi, Toshihiko Yanase, Makoto Iwayama, Qinghua Sun, and Yoshiki Niwa. 2015. End-to-end argument generation system in debating. In *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, pages 109–114, Beijing, China. Association for Computational Linguistics and The Asian Federation of Natural Language Processing.

Chenhao Tan, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2016. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proceedings of WWW*.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. Dialogpt: Large-scale generative pre-training for conversational response generation.

Style	**Example**
Raw	*It is not my job to take care of drunk friends*
What do you think	*It is not my job to take care of drunk friends, what do you think?*
What's your view	*It is not my job to take care of drunk friends, what's your view?*
Why	*Why is it that it is not my job to take care of drunk friends?*

Table 1: Input Prompts

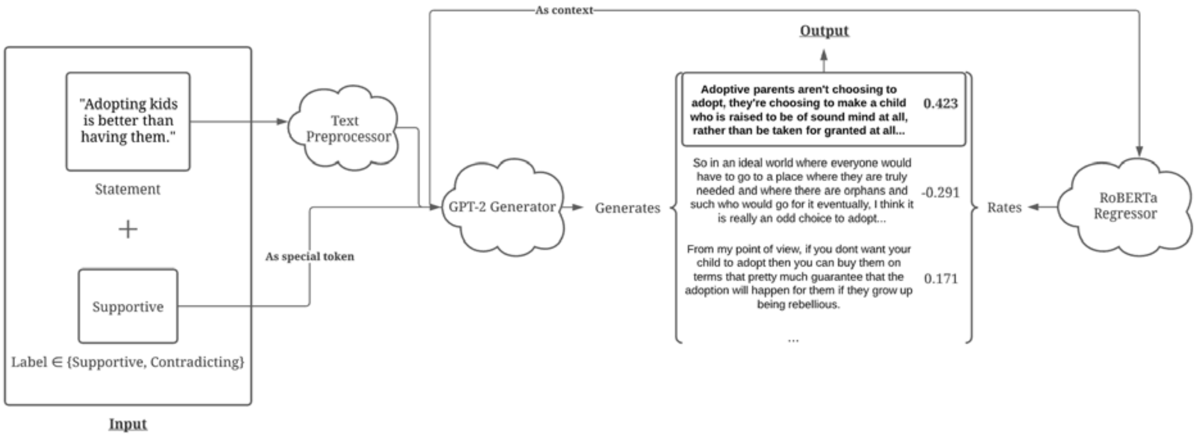

As context
Output
"Adopting kids is better than having them."
Statement
+
Supportive
Label ∈ {Supportive, Contradicting}
Input
Text Preprocessor
As special token
GPT-2 Generator
Generates
Adoptive parents aren't choosing to adopt, they're choosing to make a child who is raised to be of sound mind at all, rather than be taken for granted at all...
0.423
So in an ideal world where everyone would have to go to a place where they are truly needed and where there are orphans and such who would go for it eventually, I think it is really an odd choice to adopt...
-0.291
From my point of view, if you dont want your child to adopt then you can buy them on terms that pretty much guarantee that the adoption will happen for them if they grow up being rebellious.
0.171
...
Rates
RoBERTa Regressor

Figure 2: Generation Flow

Baseline	**ArgueBot**
13.5	72.1

Table 2: Average Generation Length, Number of Words

Model	**BLEU-2**	**BLEU-4**
Baseline	0.062	0.007
ArgueBot	0.086	0.013

Table 3: BLEU Score