ADRAGGAN: ADversarial training for RAtionale Generation: a GAN for moral dilemmas

Stanford CS224N Custom Project: `https://github.com/poojan-pandya/cs224-project`

Priya Khandelwal
Department of Computer Science
Stanford University
`priyak9@stanford.edu`

Kavin Anand
Department of Computer Science
Stanford University
akavin@stanford.edu

Poojan Pandya
Department of Computer Science
Stanford University
`poojanp@stanford.edu`

Abstract

r/AmITheAsshole: if a user has an argument or problem with morally ambiguous actors, they post on the forum with one question in mind: "Am I the asshole?" Commenters flock to the forum to provide a verdict and rationale. But what if AI could generate the top comment? Given a Reddit AITA post, can we use adversarial training to generate an explanation for the verdict? In particular, we believe that by including the verdict in our GAN as a latent variable, we can generate better explanations. Using data scraped from the r/AmITheAsshole subreddit, we train T5-small and BART model baselines on the rationale generation task. We proceeded with BART as our baseline pre-trained generator for our GAN, and began experiments with multiple Discriminator architectures: comparing CNNs vs Transformer + MLPs, including Sentiment Teacher Forcing or not, and appending the Post alongside Comment or leaving just the comment.

1 Key Information to include

- Mentor: Yuan Gao
- External Collaborators (if you have any): None
- Sharing project: NA

2 Introduction

The r/AmITheAsshole subreddit has become a popular destination for users seeking validation for their moral dilemmas. Users post their situations and ask a simple yet nuanced question: "Am I the asshole?" Fellow users then provide their verdict and a rationale for their decision. However, with the ever-growing field of artificial intelligence, the question arises: can AI generate the top comment and provide an explanation for the verdict?

The task of generating a rationale for a given verdict is a novel one in the field of natural language processing, and it presents unique challenges. While transformer models like T5-small and BART can understand context and generate text with high fluency and coherence, the challenge is to make the output feel more like a Reddit comment. In order to achieve this, we propose using adversarial training, which involves including the verdict as a latent variable in a GAN (Generative Adversarial Network).

GANs have been shown to be effective at learning distributions of data and generating outputs that closely follow that distribution. In the case of our task, including the verdict as a latent variable can help the generator avoid generating repetitive or patterned outputs, which can be a common problem in transformer models. Since the generator learns to "beat" the discriminator, adversarial training can also help the generator produce a diverse set of outputs, which can be important for maintaining the natural language feel of the generated comments.

3 Related Work

TODO: THIS IS JUST A STEM –> MAKE IT MORE TECHNICAL AND REALLY HIGHLIGHT LIMITATIONS

3.1 Existing Approaches for Rationale Generation

–> TODO: ONLY TALK ABOUT TRANSFORMER-BASED METHODS HERE

Several works have explored generating explanations for decision-making, including in the context of AITA posts. One earlier approach proposed by Dhingra et al. (2019) used an adversarial approach to generate explanations for text classification. They used a GAN-based approach to generate explanations for a given input sentence, where the generator generates an explanation and the discriminator tries to distinguish between the generated explanation and the true explanation. Similarly, Puri et al. (2021) proposed an approach that generates explanations for AITA verdicts using BERT-based models. They showed that their approach outperformed existing methods on the AITA dataset.

3.2 GANs for Text Generation

In recent years, GAN-based approaches have been used for generating explanations in different contexts. Zhang et al. (2021) proposed a conditional GAN for generating explanations for image classification. They used the predicted class and image as the conditioning variables for the generator. Given that GANs have shown improved performance in generating diverse and interpretable text, we are motivated to use GANs for generating explanations for AITA posts.

3.3 Limitations in Existing Approaches for Rational Generation Prompt the Use of GANs

Despite the success of existing approaches, there are still limitations in generating high-quality and diverse explanations for decision-making processes. For instance, BERT-based models, although achieving state-of-the-art performance, are still prone to generating explanations that are repetitive and generic. In addition, existing approaches do not take into account the underlying distribution of the data, which may result in generating explanations that are not representative of the dataset. Therefore, our proposed GAN-based approach seeks to address these limitations by incorporating the verdict as a latent variable in our GAN-based approach. By doing so, we aim to generate diverse and interpretable explanations that are representative of the AITA dataset.

4 Data

We leveraged the Reddit API wrapper PRAW to pull content from the r/AmITheAsshole subreddit. We identified a public dataset of posts, post titles, post id's, and final verdict and additionally scraped the top comment to act as a gold standard rationale for the verdict. The verdict has four classifications: *asshole*, *not the asshole*, *everyone sucks here*, *no assholes here*. [1] The scripts for scraping comments, cleaning the dataset, and preprocessing it to fit our task were written entirely by us.

Table 1: "Am I the Asshole Dataset"

Parameters	**Original**	**Cleaned**
Size	87215	81614
Average Text Length	348	330
Average Comment Length	49	49

After removing entries with "deleted", "removed", or "moderated" top comments, we achieve the data statistics seen in Table 1. We concatenated the title with the post body, and truncated the combined string to 508 words to fall within the max input sequence length of 512. In this gap, we appended a 4 word prompt, *"Am I the asshole?"*. We padded shorter inputs to 512 with a special `<PAD>` token and used each model's respective HuggingFace tokenizer to get labels and attention masks for the model inputs.

As seen in Figure A.2.1, we preserve the categorical distribution of the verdict labels across train/validation/test split. Note a strong skew towards "NTA" verdicts than other classes. This is a known weakness of the AITA subreddit. To better train our model, we are considering data augmentation techniques to up-sample the quantity of other classes to create a more uniform distribution. Describe the dataset(s) you are using (provide references). If it's not already clear, make sure the associated task is clearly described. Being precise about the exact form of the input and output can be very useful for readers attempting to understand your work, especially if you've defined your own task.

5 Approach

5.1 Baseline Models

We fine-tuned and tested multiple different pre-trained summarizers with conditional generation on our novel rationale generation task, with the ultimate goal of selecting the best-performing fine-tuned baseline as the generator architecture for our final Generative Adversarial Network (GAN).

Since we wanted to use autoregressive models with sequence-to-sequence or transformer architectures that would learn within our resource constraints, we selected BART and T5-small. We also considered DistilBert, but it would likely generate too short of rationales. We also tried fine-tuning GPT-2, but the input formatting and model size weighed heavily on our resources.

5.1.1 Baseline: T5-Small

The T5 model is a transformer-based encoder-decoder model. The HuggingFace T5 model is pretrained on the Colossal Clean Crawled Corpus (C4) dataset for a multi-task mixture of unsupervised and supervised tasks [2].We use the conditional generation model architecture with beam search for text-to-text generation. The model takes in a tokenized text input of 512 tokens and outputs a sequence of maximum 50 tokens (to align with the average comment length), which is then decoded to English. Our T5 model uses a standard cross-entropy loss, but we also compute ROUGE scores for evaluation.

For many of the HuggingFace Seq2Seq Trainer configuration settings, we decided to use the defaults, such as a ReLU activation, dropout of 0.1, and 6 decoder layers. However, upon observing phrase repetition in the output with default settings, we additionally updated the generation configuration. We implemented strict penalties by following a simple heuristic to limit repeated n-grams and imposed a penalty on repetitions. The difference can be seen in Figure A.2.2. We fine-tune for 10 epochs (20,000 iterations each), but with early-stopping, the model ultimately ran for only 4 epochs.

5.1.2 Baseline: BART

BART is a transformer-based seq-to-seq model composed of a bidirectional encoder and autoregressive decoder. BART is known to be effective on many text generation tasks, including summarization, question-answering, and machine translation. Similar to the T5 baseline, we use the default cross-entropy loss, penalize repetition, and compute ROUGE scores for evaluation.

For our baseline, we fine-tuned a BART model initially pre-trained on the CNN/Daily Mail summarization task (`bart-large-cnn` on HuggingFace [3]). We fine-tuned BART for one epoch. While we would have liked to train for more epochs, BART is an extremely large model, and our compute resources limited us to 1 epoch (of 20,000 iterations) for our initial baseline.

Table 2: "Baseline Test Results"

Metrics	**T5 w/o Repetition Penalty**	**T5**	**BART**
ROUGE-1	17.57	17.51	23.89
ROUGE-2	2.53	2.58	3.88
ROUGEL	14.95	14.91	15.55
ROUGELSUM	16.00	16.09	20.69

5.2 Adversarial Training Approach

Since GANs are used widely in computer vision to generate realistic images from a diverse distribution, we wanted to explore the possibility of using a GAN model in our natural language task to generate better Reddit comments. GANs are typically implemented via a min-max loss function that the generator seeks to minimize and the discriminator maximize. Our loss function is typical min-max loss defined as follows:

$$\frac{1}{m}[y\log D(x^i) + (1-y)(1-\log D(G(z^i))]]$$

After training our three baseline models, we settled on `bart-base` as our ultimate generator due to its superior performance on our quantitative metrics. 2 Further details regarding baseline metrics and our choice of `bart-base` are detailed in the Experiments section.

!!TODO!! Add GAN figure to this section

As described in Figure XX, we used our baseline checkpoint as a starting point for the generator, and trained a discriminator to differentiate between training samples and our generated comments. In the following subsections, we describe specific approaches to inputs, architectures, and and training tricks that we attempted to refine our GAN model.

5.2.1 Gumbel Softmax

One issue of using GANs with text data is that transformer-based text models generate text in a sequential manner, with some softmax probability distribution over the subsequent word and an argmax operation. However, the argmax operation is non-differentiable, which is an issue when trying to back-propagate discriminator losses through the generator network. To resolve this issue, we swapped the softmax layer of our BART baseline model with Gumbel Softmax, a continuous approximation of this operation. With Gumbel Softmax, we are able to successfully backpropagate through the generator.

!!TODO!! Should we add equations here for Gumbel Softmax?

5.2.2 Choice of Discriminator

While our generator was fixed to be the BART model from our baseline, we experimented with choices for our discriminator model. Since traditional GAN networks use Convolutional Neural Networks (CNNs) for the discriminator, we initially used a CNN discriminator. However, since our generator is a transformer model and transformers are generally known to work better on natural language tasks, we also tried using a transformer-based discriminator in our model.

5.2.3 Discriminator Input

When designing the discriminator, we also experimented with different inputs to the discriminator model. Initially, we only inputted real and generated top comments to the discriminator. However, since our ultimate goal is to generate comments that both resemble Reddit comments and make sense in the context of the original post, we tried using a discriminator which has access to context by using (real post, real comment) and (real post, generated comment) tuples as input to the discriminator instead. The potential downside of this approach is that the discriminator may struggle to differentiate between these since the relative number of tokens in the post is generally greater than in the comment.

5.2.4 Sentiment Classification

Since we aim for our outputs to be sensible in the context of our input posts, we hypothesized that it might be useful for the discriminator to classify the sentiment of the input text. To facilitate a more natural, interpretable "decision-making" process for the discriminator, we use a pre-trained sentiment analysis model (texttttfiniteautomata/bertweet-base-sentiment-analysis on HuggingFace) to perform inference on the input text, and use that as input to the discriminator in addition to the generated comment to provide the discriminator with context without forcing it to learn how to interpret an entire post and comment in combination.

6 Experiments

This section contains the following.

6.1 Evaluation method

In order to compare similarity between the generated rationale and gold standard comment, we decide to use BLEU and ROUGE as our metric for validation on our withheld test set, as it is a common choice for summarization-adjacent tasks. We compute ROUGE-1 (unigram similarity), ROUGE-2 (bigram similarity), ROUGEL (average longest common subsequence similarity over individual sentences), and ROUGELSUM (longest common subsequence similarity for entire summary). Based on the original ROUGE paper [4], we primarily focus our attention on ROUGE-1 and ROUGE-L, as they tend to be more insightful for short summary-like texts. We chose the combination of BLEU and ROUGE despite both of them being n-gram similarity tests as they both complement each other, with BLEU better measuring precision and ROUGE recall. [5]
Nevertheless, BLEU and ROUGE were not the best evaluation metrics for us as we want our models to focus on logic and rationale generation rather than simple n-gram match hacking. We added the Word Mover's Distance (WMD) metric to help better compare semantic propositional content. [6]

WMD scores measure the semantic distance between two text documents based on the distance that individual words would need to "move" in order to transform one document into the other, greatly helping in capturing semantic similarity than BLEU/ROUGE which can be unnecessarily punitive if exact n-grams aren't used. Further description of the formula is described in the appendix A.2.4

6.2 Experimental details

We ran many trial experiments. For selecting the optimal baseline, we played around with several hyperparameters including learning rate, repetition penalty processor, no-repeat n-grams processor, and number of epochs in a manual grid search based on several research papers. Some final hyperparameter values are displayed in the Appendix 4. All hyperparamter configurations can be found in the training args sections of our code.
Once we landed on these for our baseline, we didn't alter the basic configurations for the GAN and Discriminator. As we are already running an experimental architecture, altering too many hyperparameters later down the line could lead to convoluted and messy results. As such, we focused on the three aspects of the Discriminator: its type (CNN/BART-BASE Transformer and MLP), inclusion of Sentiment Teacher Forcing (Y/N), and Discriminator Input (Comment/Comment+Post).

6.3 Results

We found that in many cases the Baseline BART scores competitively with the GAN based models we trained afterwards. This is a surprise, as we expected the inclusion of the Discriminator will help tune the Generator to produce better results. The simpler models, Pineapple and Grape without Sentiment Teacher Forcing, had the highest evaluated BLEU and ROUGE1 scores respectively on the test set. The more complex Orange model with Sentiment Teacher Forcing, performs best on our most important metric WMD. All of the three models that beat the BART baseline's WMD score had Sentiment Teacher Forcing, leading us to believe adding this helped the model produce better, semantically correct rationale.

Table 3: **Experimental Results**

Exp ID Type	Discriminator Teacher	Sentiment Input Forcing	Discriminator Score	BLEU Score	ROUGE1 Score	WMD	Human Evaluation
Baseline: BART	NA	N	NA	0.0089	0.1807	0.9527	
Grape	CNN	N	Comment	0.0093	**0.1852**	0.9567	
Banana	CNN	N	Comment + Post	0.0104	0.1740	0.9584	
Pineapple	BART-base + MLP	N	Comment	**0.0105**	0.1729	0.9596	
Mango	BART-base + MLP	N	Comment + Post	0.0037	0.1725	0.9543	
Pear	CNN	Y	Comment	0.0088	0.1710	0.9455	
Honeydew	CNN	Y	Comment + Post	0.0085	0.1737	0.9495	
Orange	BART-base + MLP	Y	Comment	0.0095	0.1682	**0.9403**	
Blueberry	BART-base + MLP	Y	Comment + Post	0.0045	0.1776	0.9584	

Including the Post as input in the Discriminator had a negative effect on the non-sentiment teacher forcing models and a positive effect on the other. Lower ROUGE1 scores are observed for the former case, where it appears including the Post retracts from generated output. For the Sentiment Teacher Forcing models, it obtains higher ROUGE1 scores with the Post.

There seems to be no marked distinction between using a CNN versus Transformer for the Discriminator Type. Comparing like-models, the winner on WMD flip flops with seemingly no fixed pattern. This is in line with existing research, as most Discriminators are implemented as simple linear classifiers and CNNs, and transformers haven't proven its worth with its extra resources.

7 Analysis

Based on a qualitative analysis of our results, we found that using an adversarial training approach does improve the quality of our outputs. Examples of some of these outputs are shown in Figure XX. In particular, while our baseline model often produced generic responses to our input posts, the GAN-based models often produce text more specific to the situation. Our human evaluation results seems to reflect these observations, with the average rating for the GAN-based models being higher than the ratings for our baseline models. Additionally, our models which use sentiment analysis as input to the discriminator scored higher on human evaluation than models without. With these results and our empirical observations, we can believe that our approach demonstrates the effectiveness of adversarial training as a proof-of-concept for our task. In the following subsections, we describe some of our observations about the successes and failures of our model, hypotheses for these behaviors, and potential remedies.

7.1 Strengths

In general, our model performs well on short inputs, as well as inputs that have charged language and clear outcomes. This is expected, as these kinds of inputs are naturally easier to decipher and tend to have more unanimous verdicts on the r/AmITheAsshole subreddit. Additionally, when posts have more charged language, our models with sentiment teacher forcing (Pear, Honeydew, Orange, Blueberry) tend to produce better outputs than our models without. We suspect that posts with charged language are more easily analyzed by the sentiment classifier, providing more useful input to the discriminator.

7.2 Limitations

Our model often struggles on extremely long inputs. Reddit users have a tendency to ramble, and our model often has a hard time following long stories. While transformers do a better job of capturing long-range dependencies than older methods like LSTMs, handling long inputs is a general NLP problem and could likely be addressed by longer training and/or more training data. One proposal for future work is to pre-train a model for question-answering on Reddit text, and use that model to fine-tune for our task. Training on question-answering might enable the model to better understand the language of Reddit users compared to general text corpora.

Another difficult aspect of our task is that the r/AmItheAsshole subreddit is heavily biased. To garner attention, posters will often use an inflammatory "clickbait" title. At the same time, posters often spin the story in the body of the post to make themselves seem innocent. As a result, we often find that the "verdict" predicted by our model does not agree in sentiment with the actual explanation. Part of the issue is n-gram similarity can be a poor heuristic for this analysis because two pieces of text can easily have similar n-grams but convey opposite meaning, e.g. "You're the asshole" vs "I don't think you're the asshole". While we attempted to remedy this issue with our sentiment classifier within the discriminator, we think that the discriminator model has to be more complex to truly overcome these challenges.

Your report should include *qualitative evaluation*. That is, try to understand your system (e.g. how it works, when it succeeds and when it fails) by inspecting key characteristics or outputs of your model.

8 Conclusion

Summarize the main findings of your project, and what you have learnt. Highlight your achievements, and note the primary limitations of your work. If you like, you can describe avenues for future work.

References

[1] Elle O'Brien. iterative, 2020.

[2] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2019.

[3] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *CoRR*, abs/1910.13461, 2019.

[4] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[5] Fabio Chiusano. Two minutes nlp — learn the rouge metric by examples. NLPlanet, 2022.

[6] Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 957–966, Lille, France, 07–09 Jul 2015. PMLR.

[7] Edward Ma. 2018.

A Appendix

If you wish, you can include an appendix, which should be part of the main PDF, and does not count towards the 6-8 page limit. Appendices can be useful to supply extra details, examples, figures, results, visualizations, etc., that you couldn't fit into the main paper. However, your grader *does not* have to read your appendix, and you should assume that you will be graded based on the content of the main part of your paper only.

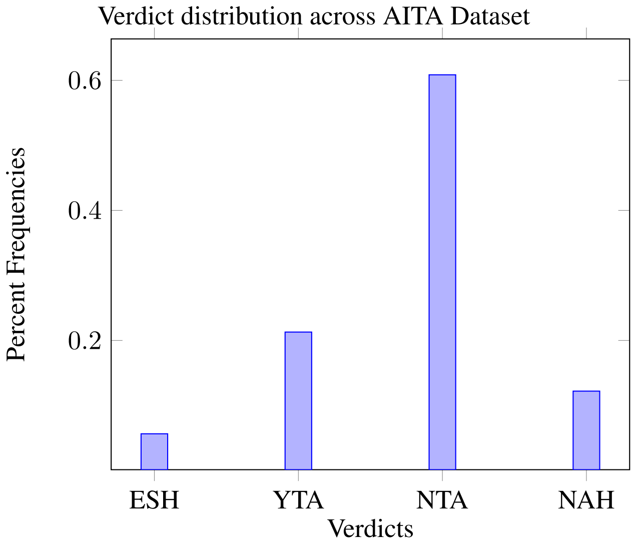
Table 4: "Hyperparameter Values"

Hyperparameter	T5 w/o Repetition Penalty	T5	BART
Learning Rate	4e-5	3e-4	3e-4
Repetition Penalty	N/A	0.3	0.3
No-Repeat N-Gram Size	N/A	4	N/A
Epoch	4	4	1

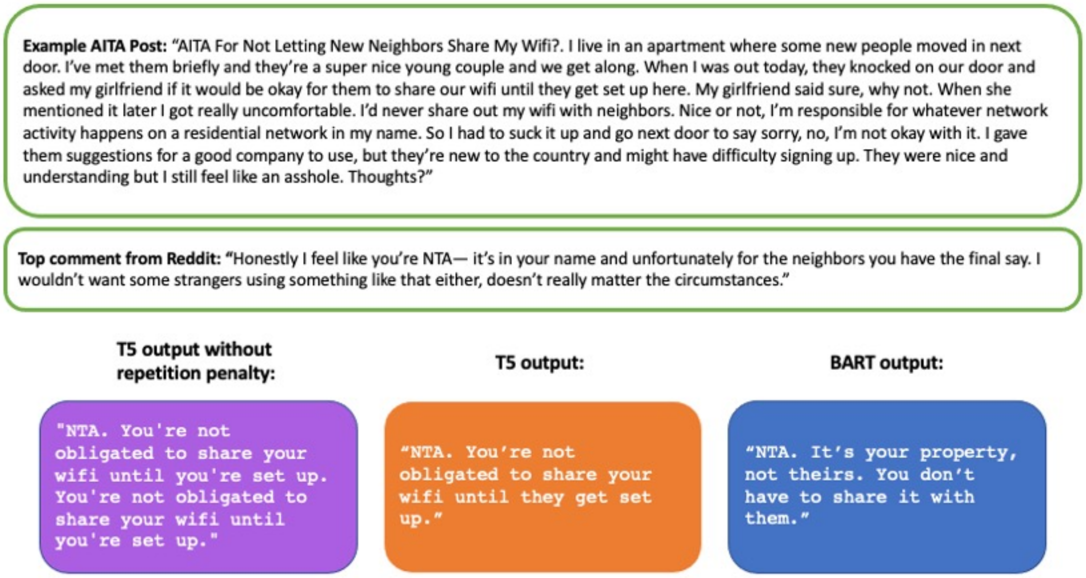
A.1 Tables

A.2 Figures

A.2.1 Dataset Verdict Distribution



A.2.2 Example Generated AITA Comment



Example AITA Post: “AITA For Not Letting New Neighbors Share My Wifi?. I live in an apartment where some new people moved in next door. I’ve met them briefly and they’re a super nice young couple and we get along. When I was out today, they knocked on our door and asked my girlfriend if it would be okay for them to share our wifi until they get set up here. My girlfriend said sure, why not. When she mentioned it later I got really uncomfortable. I’d never share out my wifi with neighbors. Nice or not, I’m responsible for whatever network activity happens on a residential network in my name. So I had to suck it up and go next door to say sorry, no, I’m not okay with it. I gave them suggestions for a good company to use, but they’re new to the country and might have difficulty signing up. They were nice and understanding but I still feel like an asshole. Thoughts?”

Top comment from Reddit: “Honestly I feel like you’re NTA— it’s in your name and unfortunately for the neighbors you have the final say. I wouldn’t want some strangers using something like that either, doesn’t really matter the circumstances.”

T5 output without repetition penalty:

"NTA. You're not obligated to share your wifi until you're set up. You're not obligated to share your wifi until you're set up."

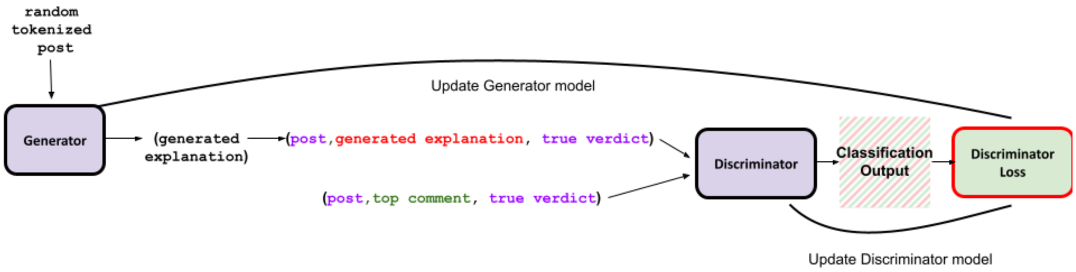
T5 output:

“NTA. You’re not obligated to share your wifi until they get set up.”

BART output:

“NTA. It’s your property, not theirs. You don’t have to share it with them.”

A.2.3 Main Approach: GAN Pipeline



A.2.4 WMD Metric

$$\mathit{WMD} = \sum_{i,j=1}^{n} \mathbf{T}_{ij} c(i,j)$$

T is a helper function that is built from the Earth Mover's Distance formula which solves the "text transportation problem." [7]

$$EMD(P,Q) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij} d_{ij}}{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}}$$