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

Stanford CS224N Custom Project: https://github.com/poojan-pandya/cs224-project
Mentor: Yuan Gao, External Collaborators: None, Sharing project: NA

**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 their dilemma on this Reddit forum with one question in mind: "Am I the asshole?" Commenters flock to the forum, reading the morally ambiguous situation, and provide a verdict and rationale on who was at fault here. The best comments that provide the most agreed upon reasoning get voted by other users into the acclaimed top comment position. Here, we explore the use of a Generative Adversarial Network (GAN) architecture to train transformer architectures to generate top comments for Reddit Am I the Asshole (AITA) posts. To determine the optimal architecture for the generator, we trained four baseline models (T5 with and without repetition penalty, BART large, and BART base), eventually deciding on BART base. Subsequently, we explored various designs for our discriminator, including Convolutional Neural Network (CNN) / Transformer architectures, differing inputs, and a novel "Sentiment Teacher Forcing" (STF) method. Upon performing quantitative evaluations using BLEU, ROUGE, and Word Mover's Distance (WMD), we found that our GAN approach leads to modest improvements in each of these metrics. More crucially for our task, we also performed a human evaluation study and found that our novel approach leads to more coherent, diverse, and sensible outputs which more accurately resemble the style of Reddit comments.

## 1 Introduction

The r/AmITheAsshole subreddit is a popular platform where users seek validation for their moral dilemmas by asking the question "Am I the asshole?" The question remains, can AI generate the top comment and provide an explanation for the verdict? This unique problem expands the role of AI beyond mere summarization and Q+A tasks and challenges the system to effectively classify a post and justify its decision [1]. While traditional transformer models can understand context and generate text fluently, they struggle with interpreting nuance in long text inputs and fall into an n-gram repetition cycle–which would make it difficult to achieve a "Reddit feel" on a comment.

GANs have been shown to be effective at learning distributions of data and generating outputs that closely follow that distribution. So, to achieve higher fluency, we propose using an adversarial training approach on transformers for text generation. Since a generator has to learn to "beat" the discriminator, adversarial training can also help the text model produce a diverse set of outputs rather

than repetitive or patterned outputs, which can be a common problem in transformer models. Our main innovations lie in adding Sentiment Teacher Forcing (STF) to our discriminators described further in 5.2.4: by including the verdict as a latent variable, the generator can handle the connotation of its outputs with more nuance and maintain a "natural language feel."

## 2 Related Work

### 2.1 Previous approaches to AITA verdict classification

While our task of generating both verdicts and rationales for data from the r/AmItheAsshole subreddit is unique, previous work has tried to perform binary verdict classification on input text. In particular, O'Brien (2020) [2] used a vanilla BERT model for binary verdict classification, but was only able to achieve 62% accuracy. This suggests that AITA content is difficult to analyze, and there is significant room for improvement.

### 2.2 Existing Approaches for Rationale Generation

A large part of our task is training our model to generate accurate explanations which agree with its verdict sentiment. Our approach to this task is partly inspired by Bacco et al (2021) [3], who use a transformer-based architecture to perform sentiment classification, while simultaneously using attention weights from their model to generate an explanation for the classification. We sought to apply this general concept to our work, in which we hope that generating explanations can help the model make more rational decisions. Furthermore, our task presents a unique challenge in that our model inherently comments on moral dilemmas, which are usually subjective and cannot be objectively answered by a language model. Talat et al. 2022 [4] describes an ethical critique of using natural language models to answer moral dilemmas, suggesting that outputs from language models are vulnerable to ethical biases in the training data. With this in mind, we hope to make clear that our model is not intended to offer a judgement on a moral dilemma, differentiating right from wrong. Rather, we simply seek to mimic and predict the style of a Reddit comment, which is useful from the perspective of academic exploration but not for impacting one's life decisions.

### 2.3 Existing Approaches to GANs for Text Generation

While GANs are popular in computer vision for generating realistic images from noise, we applied this concept to our NLP task with the hope of generating more realistic Reddit comments than the baseline. Our approach to the text GAN model was inspired by de Rosa et al. [5], who describe a way that image GANs can be modified to support text input . In particular, based on this paper, we decided to use Gumbel-Softmax in our architecture. Furthermore, Huang et al. [6] describes the effectiveness of using latent variables as part of text GAN architectures. Specifically, this paper uses sentence length as a latent variable in their sentence editing task. Based on this work, we were inspired to use generated sentiment classification as a latent variable in our model. Lastly, Rao et al. (2019) [7] uses GANs for question-answering. The approach in this paper inspired us to explore the inclusion of post context as input to our discriminator.

### 2.4 Our Contributions

While both rationale generation with transformers and GANs in NLP have been explored before with varying degrees of success, our work seeks to combine these two ideas to develop an effective NLP system for our task. To our knowledge, these ideas have not previously been explored in tandem. Within the larger conversation, we intend for our work to serve as a proof-of-concept demonstrating the potential for success when using adversarial training as an approach to rationale generation.

## 3 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:

*You're The Asshole*, *Not The Asshole*, *Everyone Sucks Here*, *No Assholes Here*. [8] The scripts for scraping comments, cleaning the dataset, and preprocessing 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 "deleted," "removed," or "moderated" top comments, we analyzed data stats in Table 1. We concatenated titles with post bodies and truncated to 508 words. We then appended a 4 word prompt to the post, *"Am I the asshole?"*, as this sort of "prompt engineering" is known to produce better results for Transformers, and can assist in fine-tuning. This fit the 512 max input length. Truncation affected 19.35% of samples, which were padded with <PAD> tokens. Figure A.2 shows preserved categorical distribution of verdict labels in train/validation/test split (98,1,1) with a skew towards "NTA" verdicts. We held out two datasets of 998 samples each to validate and test, sampling to maintain a similar verdict class distribution. To mitigate the NTA skew, we hope STF will provide the GAN with latent class information to encourage rationale for the rarer ESH and NAH posts.

# 4 Approach

## 4.1 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.

### 4.1.1 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 [9].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 other metrices for evaluation, described in section 6.1.

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 3. We fine-tune for 10 epochs (20,000 iterations each), but with early-stopping, the model ultimately ran for only 4 epochs.

### 4.1.2 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 various evaluation metrics.

For our baseline, we fine-tuned both a BART-large and BART-base model (factor of two difference in number of layers) initially pre-trained on the CNN/Daily Mail summarization task (`bart-large-cnn` on HuggingFace [10]). 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 baselines.

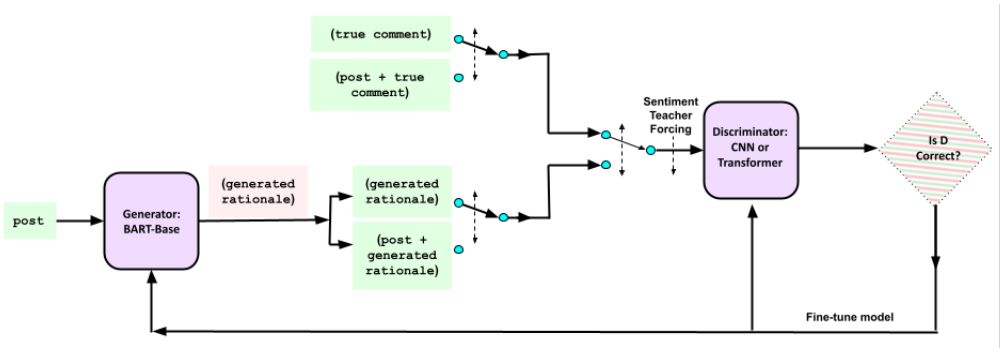



Figure 1: ADRAGGAN Model Architecture

### 4.2 Adversarial Training Approach

Our rationale generation task is an interesting subset of NLP with very limited prior papers. We enter this discussion by proposing a novel adversarial training approach on a Seq2Seq model for the rationale generation task.

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:

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

The GAN uses this loss to backpropogate on both the generator and discriminator models and update their parameters.

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

As described in Figure 1, we used our baseline checkpoint as a starting point for the generator, and trained a discriminator to differentiate between true top comments and our generated comments. In the following subsections, we describe different variations of our GAN as well as training tricks that we attempted.

#### 4.2.1 Gumbel Softmax

Using GANs with text data is problematic because transformer-based text models generate text sequentially, with a non-differentiable argmax operation. To overcome this issue, Gumbel Softmax, a continuous approximation of the argmax operation, can be used in place of softmax to allow successful backpropagation through the generator. [11] More details are provided in Appendix A.4.

#### 4.2.2 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. The two architectures are summarized in figure 2.

#### 4.2.3 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

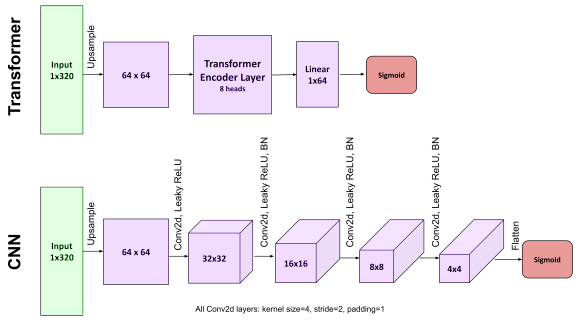



Figure 2: Comparing Transformer + MLP Discriminator Architecture vs CNN

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.

#### 4.2.4 Sentiment classification – Teacher Forcing (STF)

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 receive a sentiment score of the post itself when processing its input. To facilitate a more natural, interpretable "decision-making" process for the discriminator, we use a pre-trained sentiment analysis model (`finiteautomata/bertweet-base-sentiment-analysis` on HuggingFace) to perform inference on the post, and use that as input to the discriminator in addition to the generated comment (or generated comment with post) to provide the discriminator with context without forcing it to learn how to interpret an entire post and comment in combination.

## 5 Experiments

### 5.1 Evaluation methods

In order to compare similarity between the generated rationale and gold standard comment, we decide to use a total of four metrics, two of which denote n-gram similarity, and the other two sentiment similarity. Three are quantitative, and one is qualitative.

#### 5.1.1 N-Gram Metrics

We use BLEU and ROUGE scores as two metrics 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 [12], 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 to measure both 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 better measuring recall. [13]
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. Additionally, many "YTA" and "NTA" comments can be worded similarly, but convey entirely different meanings.

#### 5.1.2 Semantic Evaluation

To improve semantic comparison, we added the Word Mover's Distance (WMD) metric [14]. WMD measures 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. The lower the distance, the better the semantic similarity.

We also included a human evaluation metric for our 12 models, generating 10 sample outputs for each and asking two evaluators to rate the comment's potential as a top comment on a 0.0-5.0 scale. 24 evaluators graded 10 samples each, and the results are shown in tables below. More details on WMD and the human evaluation method can be found in the appendixA.3.

### 5.2 Experimental details

To design our ultimate GAN architecture, we conducted experiments in two stages. First, we trained and evaluated four baselines to determine the optimal generator for our task. Then, we attempted various approaches to the adversarial training structure and discriminator architecture to produce the best possible outputs.

In selecting the optimal baseline, we tuned several hyperparameters including learning rate, repetition penalty processor, no-repeat n-grams processor, and number of epochs in a manual grid search, based on common default values used for these models. 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 altering too many hyperparameters without enough controls might lead to convoluted and messy results. As such, we focused on the three aspects of the Discriminator to generate our eight GAN experiments: its architecture (CNN/Transformer), inclusion of STF (Y/N), and Discriminator Input (Comment/Comment+Post).

### 5.3 Results

Table 2: **Baseline Results**

| Baseline | BLEU | ROUGE1 | WMD | Human Evaluation |
|---|---|---|---|---|
| T5 w/o rep penalty | 0.0091 | 0.1731 | 0.9513 | 3.470 |
| T5 w/ rep penalty | 0.0043 | 0.1740 | 0.9584 | 3.385 |
| BART-large | **0.0094** | 0.1783 | 0.9596 | 3.545 |
| BART-base | 0.0092 | **0.1799** | **0.9488** | **3.660** |

Table 3: **Experimental Results**

| Exp ID | Discriminator Type | Sentiment Teacher Forcing (STF) | Discriminator Input | BLEU Score | ROUGE1 Score | WMD Score | Human Evaluation |
|---|---|---|---|---|---|---|---|
| Grape | CNN | N | Comment | 0.0093 | **0.1852** | 0.9567 | 4.450 |
| Banana | CNN | N | Comment + Post | 0.0104 | 0.1740 | 0.9584 | 4.195 |
| Pineapple | Transformer | N | Comment | **0.0105** | 0.1729 | 0.9596 | 4.330 |
| Mango | Transformer | N | Comment + Post | 0.0037 | 0.1725 | 0.9543 | 4.185 |
| Pear | CNN | Y | Comment | 0.0088 | 0.1710 | 0.9455 | 4.475 |
| Honeydew | CNN | Y | Comment + Post | 0.0085 | 0.1737 | 0.9495 | 4.345 |
| Orange | Transformer | Y | Comment | 0.0095 | 0.1682 | **0.9403** | **4.560** |
| Blueberry | Transformer | Y | Comment + Post | 0.0045 | 0.1776 | 0.9584 | 4.305 |

As presented in table 3 and table 4, we see that our baseline results are objectively good, and our GAN experiments show modest improvements over the baseline in each of the three quantitative metrics (BLEU, ROUGE1, WMD). Of the experimental models we tried, we found that Pineapple has the best BLEU score, Grape has the best ROUGE1 score, and Orange has the best WMD score. One promising result is that all of the three models that beat the BART baseline's WMD score had STF, leading us to believe adding this helped the model produce better, semantically correct rationales. Since WMD factors in word meaning, we consider this our most important quantitative metric. Furthermore, our results show that including the post as input in the discriminator does not seem to have a significant impact on the results. In fact, in most of our model configurations, including the post as input had a negative effect on WMD. We suspect that using the post as input to

the generator likely makes it difficult for the generator to differentiate between real and generated inputs because many of the tokens are similar. Finally, there seems to be no marked distinction between using a CNN versus Transformer for the Discriminator Type. Comparing like-models, there is no clear pattern in the numbers. Since we only used a single transformer layer, we hypothesize that using a larger transformer architecture might enhance these results, but our compute resources limited us to a smaller model.

Human evaluation is where we see the largest improvement from the baseline to GAN without STF to GAN with STF. Since word similarity is an imperfect heuristic for our task, we did not expect significant improvements to these quantitative metrics after adversarial training. As such, we are primarily focused on human evaluation, which is discussed further in the Analysis section. Average scores from each cluster of four models move from 3.52 to 4.29 to 4.42, respectively. This makes sense after analyzing our data qualitatively. Although our baseline generated comments were typically fine, they had major issues of rambling, repeating n-grams and being generally off-the-mark on the main issues. The GANs solved much of this rambling and repetition problem and, with STF, it was able to produce more semantically correct answers that were pertinent to the post.

## 6 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 3. 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. Looking at the samples outputs in Figure 3, we see that Pinapple and Orange, both of which used STF, tend to display a more diverse vocabulary such as "fat man salad" which was not a term used in the input post and "dick move", which is something a Reddit user certainly might say. We were particularly drawn to this test sample, because not only is the "NAH" top comment rare, but because the post is structured much like clickbait. The title suggests that the post warrants a "YTA" verdict, but the situation is tricky and the post's original commenters were equally split between "NAH", "YTA", "ESH", and "NTA", which is also quite rare. So, we wanted to see why our model came up with the verdict it did. While the "verdict" on these outputs (YTA) does not match the ground truth verdict (NAH), our model is producing quite opinionated results and has solid reasoning, which accurately reflects the style of Reddit comments. We are pleased with these resuls because the generated verdict does agree in meaning and sentiment with the generated explanation. 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 our observations about the successes and failures of our model, hypotheses for these behaviors, and potential remedies.

### 6.1 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 STF (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.

### 6.2 Limitations

Our model often struggles on extremely long inputs. Reddit users have a tendency to ramble, and our model sometimes struggles following long stories. While transformers are generally designed to handle long-range dependencies, it is still an issue in NLP 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.

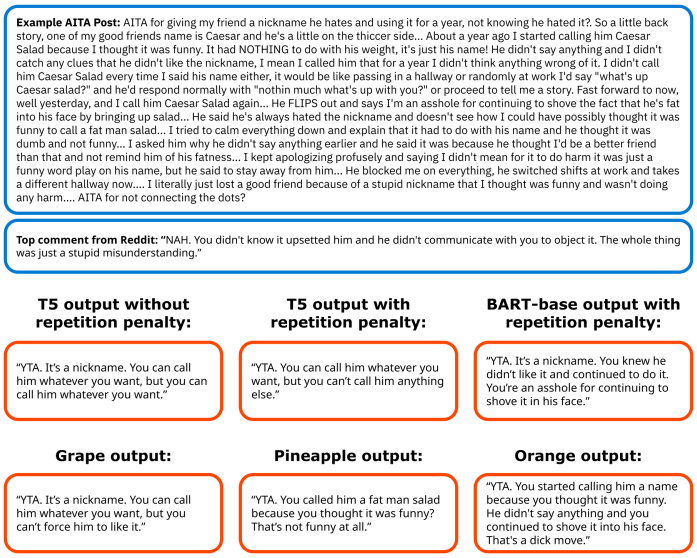



Figure 3: Qualitative analysis across baselines and leading experiments

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.

## 7 Conclusion

A GAN produced better output than a simple baseline in both quantitative and qualitative metrics. Our primary achievement, the Sentiment Teacher Forcing approach, produced the best WMD results and human evaluation metrics. Models such as Orange, that use a transformer discriminator, STF, and just the comment, produced its best results short inputs with charged language and clear outcomes. We hope to address some of the limitations we identified by further pre-training on a Reddit specific dataset and including longer samples rather than cutting input sequences to 512 words.

In future works, we'd like to explore more of Sentiment enforcing. Perhaps adding an extra classifier in the generator end of training to mark the likely verdict of the post before feeding that information to the generator will help it specialize just in rationale and produce better outputs. This could be trained in conjunction with the STF aspect of the Discriminator which could use the true verdict and comment vs classifier verdict and transformer comment in its pipeline. Another simpler avenue to explore would be obtaining a higher quality dataset with longer inputs from Reddit to pre-train more extensively on as that's where we saw massive improvements in ROUGE1 scores.

Overall, we believe that adversarial training is a promising approach to rationale generation and should be further explored.

## References


[1] Upol Ehsan, Pradyumna Tambwekar, Larry Chan, Brent Harrison, and Mark Riedl. Automated rationale generation: A technique for explainable ai and its effects on human perceptions, 2019.

[2] Elle O'Brien. Aita for making this? a public dataset of reddit posts about moral dilemmas.

[3] Luca Bacco, Andrea Cimino, Felice Dell'Orletta, and Mario Merone. Extractive summarization for explainable sentiment analysis using transformers. In *Sixth International Workshop on eXplainable SENTIment Mining and EmotioN deTection*, 2021.

[4] Zeerak Talat, Hagen Blix, Josef Valvoda, Maya Indira Ganesh, Ryan Cotterell, and Adina Williams. On the machine learning of ethical judgments from natural language. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 769–779, Seattle, United States, July 2022. Association for Computational Linguistics.

[5] Gustavo H. de Rosa and João P. Papa. A survey on text generation using generative adversarial networks. *Pattern Recognition*, 119:108098, 2021.

[6] Fei Huang, Jian Guan, Pei Ke, Qihan Guo, Xiaoyan Zhu, and Minlie Huang. A text {gan} for language generation with non-autoregressive generator, 2021.

[7] Sudha Rao and Hal Daumé III au2. Answer-based adversarial training for generating clarification questions, 2019.

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

[9] 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.

[10] 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.

[11] Harshit Dwivedi. Understanding gan loss functions. neptune.ai, 2023.

[12] 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.

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

[14] 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.

[15] Edward Ma. Word distance between word embeddings. Towards Data Science, 2018.

[16] Wanshun Wong. What is gumbel-softmax? Towards Data Science, 2020.

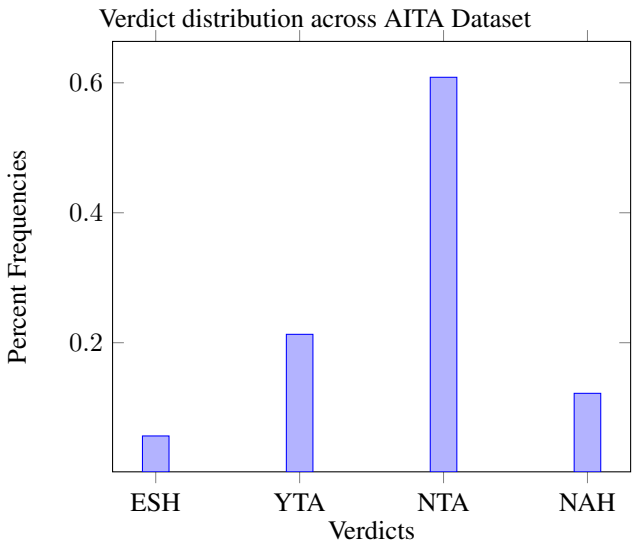
Table 4: "Hyperparameter Values"

| **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 Appendix

## A.1 Baseline Hyperparameter Settings

## A.2 Dataset Verdict Distribution




## A.3 WMD Metric

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

**T** is a helper function that is built from the Earth Mover's Distance formula which solves the "text transportation problem." [15]

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

## A.4 Gumbel Softmax

The reparameterization trick is described as such: using some independent noise as a fixed distribution, refactoring Z into a deterministic function of the parameters. Z is the categorical variable over a set of $\pi_i$ class probabilities.

$$Z = onehot(argmax_i\{G_i + \log \pi_i\})$$

We then use softmax as a differentiable approximation to argmax. This is made possible by the above reparameterization trick where we can now do backprop as its easy to compute the gradient with respect to the parameters of a deterministic function rather than parameters of the raw distribution.

$$y_i = \frac{\exp((G_i + \log \pi_i)/\tau)}{\sum_j exp((G_j + \log \pi_j)/\tau}$$

$\tau$ is the temperature parameter that marks how closely new samples approximate the discrete one-hot vectors. Low $\tau$ means computation smoothly approaches argmax, and high $\tau$ means sample vectors become uniform. [16]