Classifying and Interpreting Moral Judgment Using Reddit Data from r/AmItheAsshole

Anonymous Author(s)
Affiliation
Address
email

Abstract

This study aims to understand how moral judgments are formed within online communities, specifically through the analysis of the r/AmITheAsshole subreddit. This subreddit provides a platform where users seek judgment on whether they were morally right or wrong in various situations. By leveraging a dataset of posts from 2023, containing labels such as "You're the Asshole (YTA)" or "Not the Asshole (NTA)," we have developed models to predict these judgments based solely on the post's text. We utilized machine learning models, including BERT and LightGBM, along with topic classification via BERTopic, to identify key features influencing moral judgment. The results indicated that demographic information like gender and age, as well as the sentiment conveyed in the title, play a significant role in the community's verdict. Despite achieving higher accuracy compared to prior studies, challenges remain in fully interpreting the underlying reasons for these judgments. Future work will aim to refine the interpretability of the models and improve prediction accuracy by incorporating more nuanced features related to the post's author and relationships.

1 Introduction

As moral actors, it is common that we must ask ourselves if we are acting in ways that most other people would deem reasonable or moral. Ordinarily, one would evaluate such a thing by recalling past experiences, asking friends and family, or consulting popular media. Yet with the rise of the internet and online forums, a new method of assessing the reasonability of one's behavior has become commonplace; posting online a description of one's behavior so that other individuals may decide if their behavior is acceptable or not.

This is the exact purpose of r/AmITheAsshole, a subreddit where Reddit users can post descriptions of real-world scenarios in which they seek to understand if they were the asshole in a given situation. After the original poster (OP) makes a post, other users leave comments on the post, where they begin their messages with one of four rulings: You're the asshole (YTA), Not the asshole (NTA), Everyone Sucks Here (ESH), or No assholes here (NAH). If a post receives enough engagement (e.g., comments, upvotes, downvotes), the highest upvoted response's judgment will be assigned as the judgment of the post.

This subreddit thus allows for a novel analysis of how online communities judge behavior. Using this data, we explore the possibility of developing an AI that can evaluate morality by learning from it. Additionally, we want to examine the kinds of patterns the AI might learn from the data and what potential biases could arise when using the AI to make moral judgments.

Prior research on using Reddit data to assess moral judgments can be divided into two main categories. The first involves analyses that incorporate other users' comments on the post or actions, such as votes, in addition to the original post Botzer et al. [2022]. These models achieve high accuracy, with

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

some reaching as much as 90%. However, this kind of analysis does not allow us to understand exactly which aspects of a post lead to different judgments, and would be influenced by how the users reacted to the post rather than the post itself. Our focus is on how an AI would judge morality when given a certain situation, so prior work using comments does not meet our requirements. The second category consists of studies that analyze only the content of the original post Efstathiadis et al. [2022], Haworth et al. [2021]. While these models are less accurate, with performance around 60%, they are more relevant to our approach. However, these studies often lack interpretability, making it difficult to gain insights into what factors influence the AI's decision-making process.

In summary, the goal of this study is to interpret how morality is judged by people by interpreting the model. To achieve this, we tried two different methods. First, we trained a BERT Devlin et al. [2018] model to classify posts as YTA or NTA to see if a machine learning model can learn moral judgment. Secondly, we created a LightGBM Ke et al. [2017] model and used SHAP Lundberg and Lee [2017] to interpret its output, with the goal of understanding what aspects of a post lead to it being labeled as YTA or NTA.

2 Methodology

Our data comes from a data dump [1] of publicly available Reddit posts from r/AmItheAsshole [2]. Specifically, we have worked with data from 2023, which includes 371,849 posts. After reading in these posts, we then filtered to look only at posts labeled as either "YTA," "NTA," "ESH," or "NAH." We recoded NAH and ESH to YTA and NTA to make this an easier binary classification problem, since we expect ESH/YTA (NTA/NAH) posts to be similar.

There is a significant class imbalance present in our data (78% NTA, 22% YTA), so we re-balanced our data when training in order to assess accuracy.

After filtering for labeled posts, we were left with about 74,658 labeled posts that we could use for model fitting. For about 3,000 of these posts, the body text was deleted at the time of scraping the data, so while we have information on the title and the existence of the post, the actual text of the post was removed. Except for our topic analysis, we kept these posts in the data and used their titles instead of the body text for training.

To interpret the model, we generated several features from the text data of the posts. We hypothesized that the topic of a post would influence the moral judgments made by the community, so we used BERTopic Grootendorst [2022] to categorize each post. These were primarily grouped into five categories: family and relationships, marriage and events, financial issues, living environment and housework, and pets. The top 10 most frequent topics are listed in Appendix A. Additionally, we believed that the tone of the text would affect readers' judgments, so we performed sentiment analysis using pysentimiento finiteautomata [2024], a transformer-based sentiment analysis library. Furthermore, we extracted and inferred the poster's demographics, such as gender, age, and demonyms, as well as basic linguistic features like the number of uppercase words and profanity words, from the post content using nltk Bird et al. [2009] and profane-words zacanger [2023]. All features are listed in Appendix B.

3 Results

3.1 Classification with BERT

We designed two methods to create the textual input for BERT. The basic method is to combine the title and the main body of a post into a single paragraph as the textual input. The second, more advanced method adds the gender, age, and sentiment scores from other parts of our work, in addition to the title and the main body, to the textual input. A sample of this input is shown in Appendix C. For the advanced method, we aimed to provide social and emotional information about the post to help improve BERT's prediction and also make the BERT model more interpretable. We present the performance metrics of our fine-tuned BERT model on the validation set with two different

[1] https://www.reddit.com/r/pushshift/comments/1akrhg3/separate_dump_files_for_the_top_40k_subreddits/
[2] https://www.reddit.com/r/AmItheAsshole/

textual inputs in Table 1. The training procedures, including data preprocessing, tokenization, and fine-tuning, are described in detail in Appendix D.

Figure 1: Performance metrics of the best BERT model checkpoint on the validation set.

	Basic	Advance
Accuracy	0.5902	0.5960
F1-Score	0.6691	0.6520
Precision	0.5562	0.5670
Recall	0.8395	0.7671
ROC AUC	0.5934	0.5982
Specificity	0.3474	0.4293

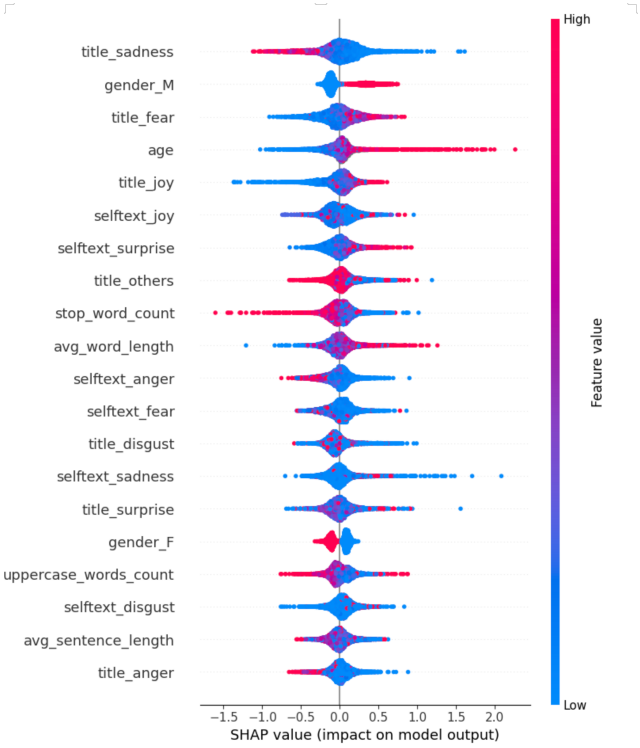

Figure 2: SHAP value for top 20 features: `gender` is encoded to one-hot vector (F for female, M for male, U for unknown). Features with the prefix `selftext_` represent sentiment expressed in the post text.

The basic BERT model achieved an F1 score of 0.6691, and the advanced BERT model achieved an F1 score of 0.6520. It is surprising to us that providing the extra features did not help improve the prediction. This supports the power of BERT to capture diverse information from pure text. However, both models outperformed the best models in Efstathiadis et al. [2022] (0.61 by BERT) and Haworth et al. [2021] (0.61 by Random Forest). Note that we only consider the models based on linguistic features from the post itself in these prior works.

Therefore, taking into account that our dataset is larger and more recent than those in Efstathiadis et al. [2022], Haworth et al. [2021], which might benefit us, we claim that our BERT model performs at least as well as the best models in these prior works.

One interesting observation about our BERT model is that it performs better on positive observations (i.e., YTA) than on negative observations (i.e., NTA), as recall is higher than specificity. One possible explanation is that there are some strong keywords in the title or the main body of YTA posts that our BERT model can easily capture. It is worth exploring this observation further in the future.

Another interesting observation is that the advanced model has a higher specificity than the basic model. This means that the advanced model makes fewer type-I errors, or false positives, than the basic model. This might be because adding the sentiment score features allows the model to capture negative sentiment more accurately. Given that calling someone an asshole is a relatively serious accusation, it might be wiser to use the advanced model. Furthermore, our main focus in building the advanced model was to make it easier for us to interpret which features might be influencing the model to predict the verdict by adding particular keywords to the model in order. Our idea was to use SHAP to see which of the keywords were generally influential. However, we need further work on this method to actually gain insights from it.

3.2 Interpretation with LightGBM and SHAP

To build a more descriptive machine learning model for predicting the verdict, we implemented a combination of the Gradient Boosting Decision Tree (GBDT) algorithm called LightGBM Ke et al. [2017] and SHAP Lundberg and Lee [2017]. Since LightGBM does not take the text itself, we trained on the features we extracted from the text and the verdict label. The results showed that combining features resulted in improved model performance. For instance, using only topic features yielded an F1-score of 0.51, whereas combining all features increased the F1-score to 0.57. However, even with all the features, the model did not perform as well as BERT, as discussed in Section 3.1, especially considering that a baseline model predicting all posts as YTA would still achieve an accuracy of 50%. This suggests that there are likely other elements influencing people's judgments that are not captured by these features.

Figure 2 illustrates the SHAP values for each feature. Higher SHAP values indicate that the feature influences the model to predict the post as YTA, while lower values indicate an influence toward predicting NTA. Features with SHAP values centered around 0 have minimal influence on the model. The color gradient represents the feature's value.

The figure clearly shows that as the emotion conveyed by the title becomes "sadder," the higher the probability of the post being judged as NTA. In other words, titles expressing sadness may influence readers to sympathize with the author. On the contrary, posts expressing fear, joy, or surprise are more likely to be judged as YTA. Interestingly, the sentiment features for titles have higher SHAP values than sentiments from the text of the post itself. This suggests that people may be biased by the title when they first see a post. Also, posts by male authors tend to receive more YTA labels, and posts authored by older individuals are also more likely to be predicted as YTA. These findings suggest that the author's emotion and role significantly influence how people perceive the post's morality, which affects whether it's labeled YTA or NTA. The author's role and their relationships with others, as described in the post, could be valuable features to include in future work. Lastly, there are also some findings that are difficult to interpret intuitively. For example, the reason why stop words or average word length had such an impact on the model's predictions is something we need to investigate further.

4 Limitations

The challenge in this task lies in creating a model that is interpretable and helps us understand how people judge the morality of a post. Transformer models like BERT often outperform decision tree-based models, but interpreting BERT's attention mechanisms is more complex than observing SHAP values. It is challenging to identify the features that influence people's moral judgments of posts and to accurately quantify and extract those features for tree-based models. Without features that reveal the content and tone of the post, the model will lack crucial information for accurate labeling. In addition, our limitations lie in the fact that the data we can gather is biased toward certain populations. Not all people will share their concerns on Reddit or answer someone's question there. Therefore, the insights we can gain from our modeling are not necessarily the general consensus of the population.

5 Future Work and Conclusions

In our work, we developed two models to classify text into YTA/NTA categories. The BERT model achieved higher scores on several accuracy metrics, including an F1-Score that was 10% better than prior works. Regarding model interpretation, we were able to interpret only the LightGBM model, yielding somewhat feasible results. SHAP values revealed that the demographics of the author, such as gender and age, are associated with moral judgment. Additionally, the sentiment of the title was more strongly associated with the judgment than the text itself. Future work involves interpreting BERT models further to gain a deeper understanding of moral judgment and exploring other models, such as Longformer Beltagy et al. [2020], to potentially increase accuracy. From this work with LightGBM, we determined that including the demographics of the author increases model accuracy. To further enhance the LightGBM model, we could incorporate features such as the role of the post's author, which might reveal additional characteristics of the author and their relationship to the people they are having issues with.

References

Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc., June 2009. ISBN 9780596516499. URL https://www.nltk.org/book/.

Nicholas Botzer, Shawn Gu, and Tim Weninger. Analysis of moral judgment on reddit. *IEEE Transactions on Computational Social Systems*, 2022.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Ion Stagkos Efstathiadis, Guilherme Paulino-Passos, and Francesca Toni. Explainable patterns for distinction and prediction of moral judgement on reddit. 2022.

finiteautomata. pysentimiento, March 2024. URL https://github.com/pysentimiento/pysentimiento.

Maarten Grootendorst. Bertopic: Neural topic modeling with a class-based tf-idf procedure, 2022.

Ethan Haworth, Ted Grover, Justin Langston, Ankush Patel, Joseph West, and Alex C. Williams. Classifying reasonability in retellings of personal events shared on social media: A preliminary case study with /r/amitheasshole. *Proceedings of the International AAAI Conference on Web and Social Media*, 15(1):1075–1079, May 2021. doi: 10.1609/icwsm.v15i1.18133. URL https://ojs.aaai.org/index.php/ICWSM/article/view/18133.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: a highly efficient gradient boosting decision tree. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 3149–3157, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *CoRR*, abs/1705.07874, 2017. URL http://arxiv.org/abs/1705.07874.

zacanger. profane-words, August 2023. URL https://github.com/zacanger/profane-words.

A Topics

Table 1: Summary information for the first 10 topics

Topic ID	Count	Topic Representation
0	30417	[parents, house, stay, later, upset, make, say...]
1	10453	[friendship, friend group, bf, upset, hurt...]
2	2662	[wedding, bridesmaids, engagement, party, bachelor...]
3	2497	[custody, pregnancy, upset, mom, relationship,...]
4	2413	[savings, financial, afford, mum, college, sibling...]
5	2226	[barking, animals, leash, neighbor, walk, stay...]
6	2157	[meals, cook, foods, dish, chicken, restaurant...]
7	1571	[cleaning, chores, laundry, dishes, roommates,...]
8	1517	[holidays, thanksgiving, family, celebrate...]
9	1276	[pets, kitten, litter box, shelter, house...]
10	1249	[roommate, pay rent, lease, house, moving...]

B Features

We built four categories of features as shown in Table2 with various methods.

For the first category linguistic, as in previous studies, the number of words and the length of sentences, as well as the number of demonyms, were extracted from the titles and body text of the posts. We used dictionaries of words for demonyms[3], stop words[4], and profanity words[5].

The gender and age features were also extracted from the posts. This is based on the fact that, in the r/AmITheAsshole subreddit, it is customary for posters to represent the gender and age of the people involved using symbols like "My bf (25M) and I (25F) are ..." Initially, we applied simple rule-based methods, achieving an initial determination rate of 58.8% for gender and 46.2% for age. For those posts where gender and age could not be initially determined, we employed the Llama 3[6] `8b-chat-hf` version. Specifically, we gave the model a prompt such as "Extract mentions of the author's gender and age from the title and body of this post." As a result, we were able to determine the gender for 64.9% of the posts and the age for 53.4%.

For the sentiment features, we employed pysentimiento[7], which is a transformer-based library for various NLP tasks. It outputs the probability of each sentiment type: anger, disgust, fear, joy, sadness, surprise, and others. Topic features are from the BERTopic analysis in the Appendix A. The model was also trained on data that contained 50% YTA and 50% NTA, and test data was also rebalanced too.

C Basic and Advanced inputs for BERT

- Basic: `AITA for not wanted babysit my niece for the entire summer? I am a SAHM (40) with 5 kids (also the oldest of my siblings). Last night my sister (37) told (but acted like she was asking in her entitled way) me I was going to babysit her daughter (7) all summer because she didn't look it to any summer camps/programs and didn't want to pay for them when I am home anyway..."`
- Advanced: `gender: Female / age: 40 / topic: parents upset stay later / title anger score: 0.0032 / title disgust score: 0.0619 / title fear score: 0.0026 / title joy score: 0.0058 / title others score: 0.9197 / title sadness score: 0.0017 / title surprise`

[3] https://github.com/porimol/countryinfo
[4] https://github.com/nltk/nltk
[5] https://github.com/zacanger/profane-words
[6] https://github.com/meta-llama/llama3
[7] https://github.com/pysentimiento/pysentimiento

Table 2: Feature names and descriptions

Category	Feature Name	Feature Description
Linguistic	title_uppercase_count	Num. of capitalizations in title
Linguistic	title_word_count	Num. of words in title
Linguistic	title_profanity_count	Num. of profane words in title
Linguistic	avg_word_length	Avg length of words in post
Linguistic	stop_word_count	Num. of stopwords in post
Linguistic	numerics_count	Num. of numbers in post
Linguistic	uppercase_words_count	Num. of capitalizations in post
Linguistic	sentence_count	Num. of sentences in post
Linguistic	avg_sentence_length	Avg num. of words per sentences
Linguistic	profanity_count	Num. of instances of profanity in post
Demography	demonyms_word_count	Num. of demonym words in post
Demography	demonyms_unique_count	Num. of unique demonym words in post
Demography	gender	Gender of post author
Demography	age	Age of post author
Sentiment	title_sentiment	Probability of each sentiment in title
Sentiment	sentiment	Probability of each sentiment in post
Topic	topic_id	Topic ID of post from Table 1
Topic	representative_words	Representative words from Table 1

```
score:  0.005 / text anger score:  0.0051 / text disgust score:
0.9394 / text fear score:  0.0033 / text joy score:  0.002 / text
others score:  0.0413 / text sadness score:  0.0071 / text surprise
score:  0.0018 / title:  AITA for not wanted babysit my niece for
the entire summer?  / text:  I am a SAHM (40) with 5 kids (also
the oldest of my siblings).  Last night my sister (37) told (but
acted like she was asking in her entitled way) me I was going to
babysit her daughter (7) all summer because she didn't look it to
any summer camps/programs and didn't want to pay for them when I am
home anyway..."
```

D Training process of BERT

For tokenization, we use the pre-trained tokenizer for `bert-base-uncased`. We pad or truncate from the end of a tokenized paragraph to make its length 512, which is the maximum possible length of a tokenized paragraph that `bert-base-uncased` can process. We padded or truncated from the end because the reader tends to read the beginning of the post more carefully and gradually lose attention or interest when approaching to the end of the post.

For the fine-tuning procedure, we list all fine-tuning hyperparameters in Table 3. The seed hyperparameter is the random seed applied to subsampling, train-val-split, and Hugging Face's trainer object. The weight decay hyperparameter is applied to Adam optimizer with weight decay fix (`AdamW` in PyTorch) Loshchilov and Hutter [2017]. The evaluation step being 200 means for every 200 iteration or batch, we evaluate the current status of our model (i.e., model checkpoint) and save it locally. At the end of the fine-tuning, we load back the model checkpoint with the best F1 score.

For the hyperparameter tuning, we tried several different combinations of hyperparameters, like learning rate, weight decay, and dropout rate. However, the F-1 scores differ within 1%. Moreover, BERT is known for being robust to hyperparameter tuning. Therefore, we did not really tune the hyperparameters and stick to the hyperparameter values shown in Table 3 that were recommended by some other work.

Table 3: Hyperparameters of the fine-tuning procedure.

Hyperparameter name	Hyperparameter value
Classifier Dropout	0.15
Learning rate	5.00E-05
Learning rate scheduler	Cosine with warmup
Batch size	16
Gradient accumulation step	4
Weight decay	2.00E-03
Epoch	5
Evaluation step	200
Evaluation metric	F1 score
Seed	547