RoFT: A Tool for Evaluating
Human Detection of Machine-Generated Text

Liam Dugan*,   Daphne Ippolito*,   Arun Kirubarajan*,   Chris Callison-Burch

University of Pennsylvania

{ldugan, daphnei, kiruba, ccb}@seas.upenn.edu

Abstract

In recent years, large neural networks for natural language generation (NLG) have made leaps and bounds in their ability to generate fluent text. However, the tasks of evaluating quality differences between NLG systems and understanding how humans perceive the generated text remain both crucial and difficult. In this system demonstration, we present Real or Fake Text (RoFT), a website that tackles both of these challenges by inviting users to try their hand at detecting machine-generated text in a variety of domains. We introduce a novel evaluation task based on detecting the boundary at which a text passage that starts off human-written transitions to being machine-generated. We show preliminary results of using RoFT to evaluate detection of machine-generated news articles.

1 Introduction

Despite considerable advancements in building natural language generation (NLG) systems that can output extremely fluent English text, there is still not very much understanding of how humans perceive such text. This understanding is crucial both to evaluate improvements in NLG systems and to analyze the societal ramifications of machine-generated text becoming increasingly easy to produce.

When evaluating NLG systems, it is considered standard practice to ask evaluators to rate generated text on criteria such as fluency, naturalness, or relevance to a prompt on a Likert scale (van der Lee et al., 2019). Preference studies, where a rater is shown two generated excerpts and asked which one they prefer, are also common. Other recent work has focused on the detection problem: how capable are humans of distinguishing textual excerpts gen-

*Authors listed alphabetically contributed equally.

Figure 1: A word cloud of common words that annotators used to describe why they thought sentences were machine-generated.

erated by a system from those written by another human (Ippolito et al., 2020; Zellers et al., 2019).

However, due to the prohibitive cost of running human evaluation tasks, much of these evaluations have been rather limited in scope. For example, such evaluations usually show results on only a single category of text (news articles, stories, webtext, etc.). This could be problematic since different domains have different levels of named entities, world facts, narrative coherence, and other properties that impact the success of NLG systems. In addition, most papers only evaluate on a very limited selection of decoding strategy hyperparameters. Holtzman et al. (2019) and Ippolito et al. (2020) both show that the decoding strategy chosen at inference time can have a significant impact on the quality of generated text.

In this work, we introduce the Real or Fake Text (RoFT) system, a novel application for simultaneously collecting quality annotations of machine-generated text while allowing the public to assess and improve their skill at detecting machine-generated text.

The system focuses on the detection task by showing annotators an article of text one sentence at a time. The first several sentences are from a real human-written text source and the next several sentences are a machine-generated continuation. The user's goal is to guess where the boundary is. When they think that a sentence is machine-generated, they are asked to give an explanation for their choice, and afterwards the true boundary is revealed.

In the remainder of this paper, we discuss why we think this task is interesting from a research perspective and describe the technical details behind our implementation. We show preliminary results that showcase the types of analyses that are possible with the collected data, and finally we discuss plans for future work.

The RoFT website is located at `http://www.roft.io/`. The source code is available under an MIT License at `https://github.com/kirubarajan/roft`.

2   Research Motivations

The purpose behind RoFT is to collect annotations on the scale needed to probe further analysis of machine-generated text.

2.1   Length Threshold for Detection

State-of-the-art generative models tend to produce text that is locally fluent but lacking in long-term structure or coherence. Intuition suggests that fluent NLG systems ought to produce text that is high quality for long durations (measured in number of sentences). As such, we are interested in the task of detecting the boundary between human-written text and machine-generated text as a comparison method for NLG systems.

2.2   Text Genre/Style

Generative language models have now been trained and fine-tuned on a great diversity of genres and styles from Tweets and short stories (Fan et al., 2018) to Wikipedia (Liu et al., 2018) and news articles (Zellers et al., 2019). Each of these datasets has its own distinct challenges for generation; for example, in the story domain it is acceptable for a generator to make up facts while this would be unacceptable in a Wikipedia article. We are interested in how these differences might impact the ability of humans to detect machine-generated text.

2.3   Why Text is Bad

van der Lee et al. (2019) demonstrated that less than 3 % of recent papers on NLG ask for free-text comments when performing human evaluations. And yet, understanding why humans think text is low quality can be very important for diagnosing problems in NLG systems (Reiter and Belz, 2009). Therefore, the RoFT platform collects free-text explanations from our annotators on their selection. Such data could provide insights into: the types of errors that NLG systems introduce, the types of errors humans are sensitive to, and even the types of errors human-written corpora contain (when a rater inadvertently predicts that a human-written sentence is machine-generated).

3   System Overview

This section gives an overview of the design of RoFT, including the task that annotators were asked to complete and gamificiation strategies used to encourage organic traffic.

3.1   Task Definition

Users first choose which category they would like to play in, where different categories correspond to different text domains or NLG systems. The annotation "game" then consists of a series of rounds. Figure 2 gives screenshots of the flow of a single round. Each round starts with the user being presented a single sentence that is guaranteed to be human-written. For example, this might be the first sentence of a New York Times article. Afterwards, users may select to display more sentences, one at a time. At each step, they must decide if they believe that the most recent sentence is still written by a human. When the user decides they are confident that a machine has written the most recent sentence (i.e. they have found the "boundary sentence"), the round ends. The user is then asked to provide a natural language explanation of what prompted their decision. In essence, the annotators' goal is to identify the exact sentence where a machine "takes over" and the text deviates from human control.

3.2   Implementation

The RoFT annotation system aims to facilitate our experimental design in a reproducible manner. In particular, our system tracks a variety of metadata about both the annotations and the generation strategy for each example. These include: the order in which a user completed annotations, the type

(a) The user is shown an initial sentence and then one sentence of continuation at a time. At each step, the user decides if the latest sentence is human-written or machine-gemerated and pressed the appropriate button.

(b) When the user decides that the most recent sentence is machine-generated, they are asked to provide an expanation for their decision.

(c) The true boundary is then revealed. In this case, the user would be alterted that they received 5 points since they guessed the boundary correctly.

Figure 2: The user interface for annotation.

Figure 3: A user's profile page.

of user account associated with each annotation (e.g. paid worker or organic traffic), and the decoding strategy used to produce each generation. The system was developed in Python using the Django Framework and a SQL database. The use of a relational database enables sophisticated queries to be made on the collected annotations for analysis. We plan to make dumps of the database available to other researchers to further promote research into the evaluation of generated text.

3.3 Gamification

Since the cost of collecting human annotations via a crowd platform such as Amazon Mechanical Turk can be prohibitively expensive for large studies, we aimed to build the RoFT website in a manner that would encourage sustained participation without the need for a financial incentive.

Each user has a Profile page (shown in Figure 3) where they can see statistics on the total number of annotations they've done, how many points they have earned, and how many questions they have answered perfectly. There is also a leaderboard where users can check how their point count compares to other raters. The leaderboard encourages users to do more annotations, since this is the only way to move up on the rankings.

We received unsolicited compliments from our initial annotators such as "Interesting, fun task" and "Really convincing passages." We intend to add further gamification elements, including leaderboards broken down by text domain, comprehensive statistics on user progress and skill, and the ability to see and up-vote the free-text comments of other users.

3.4 Generations

We ultimately plan to use RoFT to study differences in detection performance across a variety of NLG systems and text domains. The initial version of RoFT includes two complementary categories of text: news and fictional stories. Users have the option to choose which category they would like to annotate.

For the news category, prompts are drawn from the New York Times Annotated Corpus (Sandhaus, 2008) and are truncated to between 1 and 10 sentences long. GROVER (Zellers et al., 2019) is then conditioned on these starting sentences and asked to complete the article. Finally, the outputs from GROVER are truncated so that the sum total number of sentences for each example is 10.

The data on fictional stories was prepared similarly except that the Reddit Writting Prompts dataset (Fan et al., 2018) was used for the prompts, and the GPT-2 XL model (Radford et al., 2019) was used for generation.

Each category contains over 1,500 examples, where for each example the number of human-written context sentences as well as the values of the decoding strategy hyperparameters were chosen randomly. For our initial seeding of data, Nucleus sampling (Holtzman et al., 2019) was used for all decoding, where the p hyperparameter, which controls the diversity of the generated text, was randomly selected to be anywhere from $p = 0$ (argmax) to $p = 1.0$ (full random sampling).

4 Case Study

To show the efficacy of the data collected from our initial pilots, we present a case study on conducting over 3000 annotations on the RoFT platform using generations from the news article domain.

4.1 Data Collection

While our eventual hope is for the website to have enough organic traffic for useful data to be collected, for the purposes of this study, two hundred Amazon Mechanical Turk workers were paid to complete 10 annotations each on the website. In total, we collected 3244 annotations (7.9% of annotators continued past the minimum of 10 questions they were required to do to get paid). 10% of examples the crowd workers saw were designated attention check questions in which the prompt explicitly stated they should select "human-written" at every step. About 25% of crowd workers failed

this check, and after filtering out these annotators, we were left with a total of 1848 high-quality annotations, which we will refer to as the filtered annotation set.

4.2 Inter-Annotator Agreement

There were 768 examples which had at least two crowd workers provide annotations for them (645 of which had at least three annotations provided). This led to 6,115 instances of pairs of annotations on the same examples. Of these, 18.3% predicted the exact same sentence as the boundary, and 28.4%, predicted boundaries at most one sentence apart from each other. When considering only the filtered annotation set, there were 2,064 pairs of annotations. Of these, 18.6% predicted the exact same sentence as the boundary, and 28.3% predicted boundaries at most one sentence apart from each other.

4.3 Evaluation

Accuracy Among annotators that passed our attention check, 15.8% of the filtered annotations correctly identified the exact boundary between machine and generated text. Additionally, the average annotation from our filtered set was 1.989 sentences after the true boundary. This is consistent with our intuition, namely that current state-of-the-art NLG systems are capable of fooling humans but typically only for one or two sentences.

Distance from Boundary In Figure 5 we show a histogram of annotations grouped by their distance away from the boundary sentence. This distribution is far from symmetrical, with the left tail (composed of human-written sentences) modeling more closely the expected random distribution while the right tail (composed of machine-generated sentences) decreases linearly. This indicates that our annotators are successfully picking up on clues in generated text and thus the sentence-by-sentence structure of the RoFT experiment is an effective way to evaluate text. This preliminary results bodes well for the future large-scale use of the tool.

Point System While accuracy may be a simple and intuitive metric for assessing performance, it is sub-optimal for our purposes as it does not give partial credit for guesses that are after the boundary, despite such guesses being successful identifications of generated text. Average distance (in sentences) from boundary is not sufficient either, as it does not weight all guesses before the boundary equally
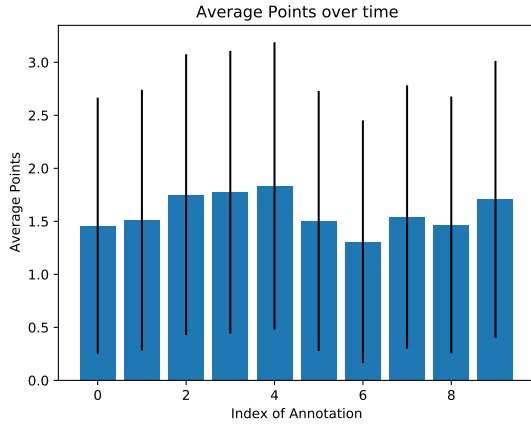
Figure 4: The average number of points (Section 4.3) received per annotation in the filtered annotation set grouped by the temporal order in which they were shown to the annotators 0 (first) to 9 (last).
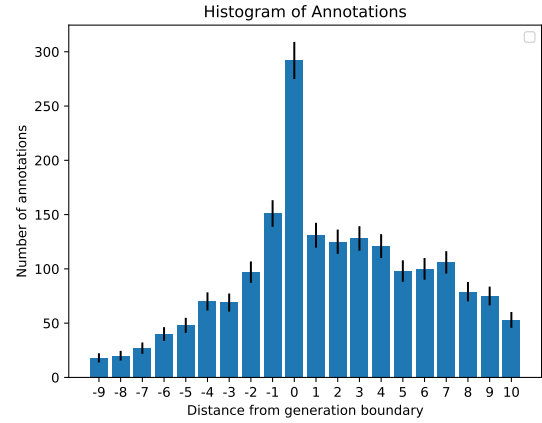

Figure 5: A Histogram of the filtered annotation set grouped by the distance (in number of sentences) between the selected sentence and the true boundary sentence

negatively and thus over-penalizes early annotations on generations with late-occurring boundary sentences.

To combat these issues, we developed a point system to properly capture a better representation of annotator ability. After each annotation, a user is assigned points based on their performance: 5 points for guessing exactly on the boundary and a linearly decreasing number of points for each sentence beyond the boundary. No points are awarded for guesses that appear before the boundary. Going off this system, we use the average points per annotation as our metric for the experiments shown in Figure 6 and Figure 4.

4.4 Skill Range of Annotators

There was a significant range in detection ability across the crowd workers. The top 5% of the filtered worker pool earned an average of 3.34 points per annotations while the bottom 5% earned an average of 0.35. Since it is difficult to separate out the influence of inherent skill from that of misaligned incentives (AMT workers were paid for completion, not correctness), more research is necessary to understand differences in annotator ability.

4.5 Impact of Decoding Strategy

During our small-scale case study, we did not see a noticeable correlation between the values of the Nucleus Sampling (Holtzman et al., 2019) hyperparameter p and the detection accuracy of humans as reported in Figure 6. This is likely due to the low number of annotations per value of p (n=180) and we hope to run a more comprehensive version

of this experiment with more data in the future.

4.6 Impact of Revealing the Boundary

As part of the gamification aspect of the RoFT platform, we reveal the true boundary to our annotators after every annotation they complete. This feature adds a level of interactivity to the process and is crucial for ensuring that the RoFT experiment is enjoyable and appeals to the general public. To better understand how this decision affects our data, we analyzed if our annotators got more accurate as their annotation session progressed in Figure 4. Our experiments show that revealing the boundary to our annotators has little to no effect on their accuracy, sometimes even lowering it. Future studies using the RoFT platform will seek to further investigate if human annotators can be trained to detect generated text over long periods of time and multiple sessions.

4.7 Free-form Comments

Our proposed annotation system allows annotators to provide a natural language explanation of why they made a particular decision (e.g. classifying a sentence as human-written or machine-generated). Due to minimal oversight, many annotators re-used or copy/pasted their comments across annotations. Filtering for duplicates, we collected over 1200 unique comments, out of around 3000 annotations. Manual inspection shows that many annotations relied on similar clues such as: problems with entailment, formatting (i.e. punctuation), and repetition. These responses can be used to inform future improvements to existing NLG systems and decoding
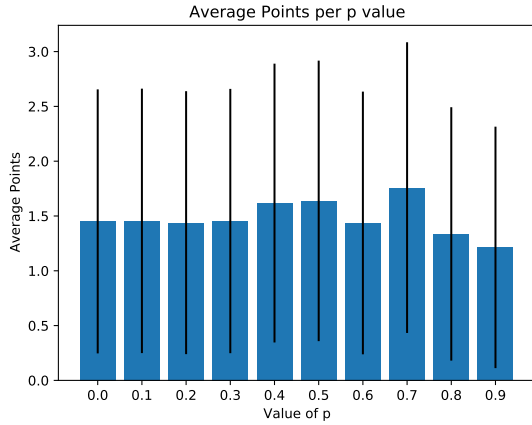
Figure 6: The average number of points (Section 4.3) received per item in the filtered annotation set grouped by the value of p used for decoding.

Sample Annotation
Seems like a conversational statement that doesnt logically follow from a book title reference
not relevant to preceding sentences
I don't think that a human would write about tarot cards in an obituary and it says obituaries plural.
The sentence is too short and simple, sweating computerized.
First time I heard of dinosaur-eating mammals
The sentence is left hanging.
Repeated the second line again and To is written as TO

Table 1: Examples of explanations crowd workers gave for why they thought a sentence was machine-generated.

strategies. Additionally, it is possible to use data mining techniques to extract an error taxonomy from the provided natural langauge description of errors.

5 Related Work

Nearly all papers in NLG do some form of human evaluation, usually using Amazon Mechanical Turk (van der Lee et al., 2019). Typically the interfaces for these evaluations are simple web forms. van der Lee et al. (2019) offers a survey of many of these methods. Custom-designed websites for collecting or displaying human evaluations of generated text have become increasingly prominent in the open-ended dialog domain, with ChatEval (Sedoc et al., 2018) and ConvAI (Pavlopoulos et al., 2019) being two examples.

However, RoFT was primarily influenced by other "real or fake" websites that attempt to gamify the detection task, such as http://www.whichfaceisreal.com/ for generated face images and https://faketrump.ai/ for generated Tweets. The task is similar to the one used for human evaluation in Ippolito et al. (2020), except in their task the text shown to raters was either entirely human-written or entirely machine-generated.

The boundary detection task we propose was inspired by the Dialog Breakdown Detection Challenge (Higashinaka et al., 2016), in which the goal is to automatically detect the first system utterance in a conversation between a human and a chatbot system that causes a dialogue breakdown.

6 Conclusion and Future Work

In this work, we have introduced RoFT and shown how it can be used to collect annotations on how well human raters can tell when an article transitions from being human-written to being machine-generated.

Ultimately, we plan to use RoFT to conduct a large-scale systematic study of the impact of decoding strategy, fine-tuning dataset, prompt genre, and other factors on the detectability of machine-generated text. We also intend to collect and release a large dataset of natural language explanations for why humans think text is machine-generated. We hope that these will provide insights into problems with both the human-written text we use as prompts and into the types of errors that NLG systems make.

Such a study will require tens of thousands of human annotations. We hope that by gamifying the annotation process and encouraging organic traffic to the website, we can ultimately bypass the need for crowd workers who, since they are paid by the annotation, are disincentivized from taking the time to provide high quality annotations.

We believe that RoFT provides a powerful tool for understanding the strengths and limitations of a great variety of NLG systems, and we look forward to working with researchers interested in testing out their own model outputs within the RoFT evaluation framework.

References

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*.

Ryuichiro Higashinaka, Kotaro Funakoshi, Yuka Kobayashi, and Michimasa Inaba. 2016. The dialogue breakdown detection challenge: Task description, datasets, and evaluation metrics. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3146–3150.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Daphne Ippolito, Daniel Duckworth, Chris Callison-Burch, and Douglas Eck. 2020. Automatic detection of generated text is easiest when humans are fooled. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1808–1822.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. 2019. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368.

Peter J Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. 2018. Generating wikipedia by summarizing long sequences. *arXiv preprint arXiv:1801.10198*.

John Pavlopoulos, Nithum Thain, Lucas Dixon, and Ion Androutsopoulos. 2019. Convai at semeval-2019 task 6: Offensive language identification and categorization with perspective and bert. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 571–576.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Ehud Reiter and Anja Belz. 2009. An investigation into the validity of some metrics for automatically evaluating natural language generation systems. *Computational Linguistics*, 35(4):529–558.

Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.

Joao Sedoc, Daphne Ippolito, Arun Kirubarajan, Jai Thirani, Lyle Ungar, and Chris Callison-Burch. 2018. Chateval: A tool for the systematic evaluation of chatbots. In *Proceedings of the Workshop on Intelligent Interactive Systems and Language Generation (2IS&NLG)*, pages 42–44.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. In *Advances in Neural Information Processing Systems*, pages 9054–9065.