Fuzzy Systems and Data Mining VI
A.J. Tallón-Ballesteros (Ed.)
© 2020 The authors and IOS Press.
This article is published online with Open Access by IOS Press and distributed under the terms of the Creative Commons Attribution Non-Commercial License 4.0 (CC BY-NC 4.0).
doi:10.3233/FAIA200717

Exploring the Potential of GPT-2 for Generating Fake Reviews of Research Papers

Alberto BARTOLI [a,1] and Eric MEDVET [a]

[a] *Department of Engineering and Architecture, University of Trieste, Italy*

Abstract. Modern tools for natural language generation may enable novel forms of scholarly fraud based on the *automatic* generation of *fake review reports* for academic papers, i.e., of a few sentences broadly related to the textual content of a submission and written with the style of an anonymous reviewer. A tool capable of generating such reports automatically and for free could enable various forms of unethical behavior by publishers and researchers. In this work we experiment with a simple heuristic that makes use of widely available and easy to use tools for natural language generation, including the *Generative Pretrained Transformer 2 (GPT-2)*, in order to craft fake reviews automatically. We also perform a small user study for assessing the credibility of those reviews. Our analysis suggests that academic frauds based on fake reviews may indeed be feasible and ready to be deployed in the wild.

Keywords. academic fraud, artificial intelligence, bibliometry, language models, natural language generation, natural language processing.

Introduction

Peer review of research papers is a cornerstone of scholarly publishing and is widely believed a crucial element for ensuring quality of published research. Peer review must be done by experts in the specific field and must be fair, accurate, and timely. Satisfying these essential requirements is becoming more and more difficult [1,2,3], which could encourage certain publishers to not perform stringent peer reviews in order to expand their customer base and attract more submissions from authors. Indeed, the incentives that drive the behaviors of the many actors involved in scholarly publishing—authors, publishing companies, conference organizers, editors, reviewers, research institutions—do not necessarily lead to an overall scientific progress and have often resulted in various forms of questionable behavior if not plain fraud.

In this work we explore the feasibility of a novel form of scholarly fraud based on the automatic generation of fake review reports for academic papers, i.e., of a few sentences with just some generic criticisms or recommendations broadly related to the textual content of a submission and written with the style of an anonymous reviewer. A tool capable of generating such reports automatically and for free could enable various forms

[1] Corresponding Author. E-mail: bartoli.alberto@units.it.

of unethical behavior by publishers and researchers. A journal or conference could return one or more fake review reports to authors in order to simulate an accurate vetting procedure, while busy researchers asked to review a paper could return fake review reports for justifying their presence in boards and committees. We are aware of only one prior attempt at generating fake reviews for scientific papers automatically, proposed by our research group [4]. The proposal was based on a template system constructed from a small set of existing reviews and specialized with terms specific of the paper being reviewed. The generated reviews were thus severely limited in the richness and diversity of text, but were considered credible by several readers (please refer to the cited work for full details). In this work we explore the usage of modern neural language models, such as the *Generative Pretrained Transformer 2 (GPT-2)* [5,6], that were not available at the time the cited proposal was developed. Indeed, recent developments in "artificial intelligence" have enabled various applications able to emulate the behavior of a human to an extent that appeared not possible just a few years ago.

1. Our Framework

1.1. Natural Language Generation

At the core of our framework there is the GPT-2 language model developed by the OpenAI research institute. GPT-2 has been constructed in an unsupervised way from a large corpus of 8 million web pages [5,6]. The training objective consisted in predicting the next word given all the previous words in the stream. The typical usage of GPT-2 consists in generating synthetic text samples in response to a textual input. The generated text follows the style and content of the input, which in principle allows generating realistic continuations about any topic inserted as input. GPT-2 is available in several versions depending on the number of parameters in the model, that range from 124 millions ("small"), to 1.5 billions ("extra large"). The two smallest versions of GPT-2 can be fine-tuned on a selected input text in order to make the text generated by the fine-tuned model "more typical" of the domain associated with that text. Naturally, given the large amount of model parameters, such a fine-tuning cannot eliminate any bias included in the original training corpus and a fairly large amount of additional text is required, in the order of several millions of characters.

We fine-tuned the "medium" version of GPT-2 (355 million parameters) with a 55 MB textual corpus of review reports. We constructed this corpus based on a dataset of peer reviews of scientific papers that has publicly made available recently[2]. This dataset, called *PeerRead*, contains reviews from several conferences in machine learning and the authors demonstrated its usage for predicting acceptance/rejection and numerical scores [7]. We extracted from PeerRead only the textual reports and discarded all the scores. We used also the reports from NIPS conferences 2018 and 2019, not included in PeerRead. We chose to fine-tune a single language model on the full set of review reports available rather than, e.g., fine-tuning a model on positive reviews and another model on negative reviews. We used the resulting language model, *GPT-2-ReviewGenerator*, as described in the next section. We performed fine-tuning and text generation based on a publicly available Colaboratory Notebook [8].

[2] https://github.com/allenai/PeerRead

1.2. Fake reviews generation

We experimented with a simple procedure for fake reviews generation, designed to minimize human intervention and, most importantly, amenable to be implemented with skills at the level of an editorial assistant. The actual implementation of our experiments required some manual steps but, as will be evident, these steps can be automated easily.

Practical usage of GPT-2 involves several parameters, in particular: p (*prefix*), the textual input for conditioning the generated text; n_{samples} the number of different textual outputs to be randomly generated (*samples*), all conditioned by the same input p; len, the desired length for the samples; $temperature$ and top_k, two numerical values which control the degree of randomness in the generation of the words for the samples (please refer to [8] for details).

Our procedure takes title and abstract of the paper as input, along with a short and possibly incomplete sentence biased toward the desired outcome for the review, e.g., “This is a solid work and should be accepted because”, “The experimental section is flawed and I cannot recommend acceptance” or something alike. The procedure is as follows.

1. Use an automatic summarization service for constructing a summary of the paper based on abstract and introduction. The length of the summary, denoted by s, should be 80–100 words.
2. If the summary contains any sentence in active voice, rephrase the sentence in passive voice, e.g., “We propose an algorithm” should become “An algorithm is proposed”.
3. Write the beginning of a short sentence biased toward the desired outcome for the review, as indicated above. Let this *driving sentence* be denoted as d.
4. Concatenate s and d and use the resulting string as input prefix p to GPT-2-ReviewGenerator. For the other parameters we used $n_{\text{samples}} = 10$, $len = 800$ words, $temperature = 0.7$ (default value) and $top_k = 40$ (default value).
5. Analyze the generated samples and choose a single, contiguous snippet that satifies these requirements: (a) it has adequate length (between 150 and 300 words); it is coherent with the desired outcome (accept vs. reject); (b) it is internally coherent (e.g., it does not provide both strongly positive and strongly negative comments); (c) it does not contain elements that might be totally unrelated to the paper being reviewed (e.g., references to prior publications, acronyms of algorithms and alike); (d) it looks natural (this requirement obviously requires human judgement). The chosen snippet constitutes the output of our procedure, i.e., the fake review report.

Regarding step 5, the choice of the snippet from the generated samples took no more than 5 minutes for each paper. Several samples could be discarded immediately because after a more or less creative beginning, the generated text started to repeat itself. The requirements at step 5 could be modified or extended in several ways. We do not elaborate on this point for brevity.

All the steps of the procedure could be automated, either in full or in part, with the only exception of step 5-(d). The automatic summarization at step 1 can be done with one of the many techniques and services for automatic summarization that exist. According to our early experimentation, the relative length of abstract, introduction, and summary makes the summary not very dependent on the specific summarization technique used.

We chose the target length for a review report (i.e., requirement (a) above) based on the length of the many thousands of reports in our (augmented) PeerRead dataset: for two of the conferences of the dataset, it is reported a mean of 531 and 346 words and a standard deviation of 323 and 213 words. While a longer report could be, in principle, more credible, obtaining a long, coherent text from GPT-2 tends to be difficult.

We emphasize that we strive for simplicity, in particular, with respect to the actions required by human operators. In this respect, we remark that we select a *single* snippet as review report, rather than a text obtained from the samples in a more elaborated way, e.g., by merging a number of different snippets. For example, it would be easy to greatly improve the "quality" of generated reviews by simply concatenating a few selected snippets, even without any modification to those snippets. We preferred to not explore more elaborated options, not only for keeping the procedure as simple as possible but also for a better assessment of the power of modern tools for natural language generation in our domain of interest.

2. Assessment

Several metrics exist for assessing the quality of a natural language generator tool, e.g., lexical richness, syntactic complexity, complexity, and diversity [9,10,11]. In our opinion such metrics are not adequate for our fake reviews tool: they would assess more the GPT-2 (fine-tuned) model than the fake review generator; and, they would not address the real issue of the credibility and usefulness of the generated reviews. It would be necessary to insert those reviews in a real reviewing process and verify their actual impact, which would depend on the target of the fraud. A busy researcher asked to review a paper could return a fake review, in which case we should assess whether the journal editor or conference program committee are actually able to detect the fraud. Conversely, and perhaps most interestingly, a journal or conference could return one or more fake reviews to authors in order to simulate an accurate vetting procedure, in which case we should assess whether the fraud may be detected by authors (assuming that authors are indeed interested in having their papers actually reviewed, rather than only in having their papers published). We are not able to perform this kind of assessment and thus performed a small user study, described below.

For our assessment we selected three papers published by our research group [12, 13,14], in venues different from those of the dataset that we used for fine-tuning GPT-2. For each paper we selected a real review that recommended acceptance and generated two fake reviews, one recommending acceptance, and another recommending rejection. All the real reviews were, in our opinion, of good quality. The resulting set of 9 reviews is available on line, in the "Data and tools" section of the web site of our laboratory[3]. We provide two fake reviews in Table 1 for completeness.

Then, we constructed a form containing, for each paper, title, abstract, and the three reviews. Each review consisted of the recommendation (accept vs. reject) and the textual report. The order of the three kinds of reviews was different for each paper. The form asked to rate each report as "Very useful", "Useful", "Useless", "Completely useless". We circulated the form between some of our academic contacts, by asking collaboration for a generic "research on the quality of peer reviews" and *without* specifying that some

[3] https://machinelearning.inginf.units.it/

Regex golf has recently emerged as a specific kind of code golf, i.e., unstructured and informal programming competitions aimed at writing the shortest code solving a particular problem. A problem in regex golf usually consists in writing the shortest regular expression which matches all the strings in a given list and does not match any of the strings in another given list. In this paper, a methodology based on Genetic Programming for generating solutions to regex golf problems is proposed—a regex golf player. A population of candidate regular expressions represented as trees and constructed with carefully selected regular expression operators and constants is generated. This paper is solid work that should definitely be accepted because it shows an ability to generate solutions for the regex golf problem. The authors have done a good job in developing the scheme and the general idea of the paper. The authors have also shown a clear connection between the concept of Genetic Programming and the solution to the regex golf problem. I think this is a good result. The paper is written well and the authors have done a good job in developing their scheme. However, some issues remain. The paper is not clear about the significance of the tree structure. Does it reflect the fact that the tree structure is only used to encode the solutions to the problem? Is it the case that this structure is only used to encode the solutions to the problem? If so, then it is unclear what happens to the solution when a new tree is created? What happens to the solution when the current tree is deleted?
The experimental section is unclear because the paper does not present any data in the experiment. The proposed approach is difficult to evaluate, as the paper presents results only when an operator is used and does not describe the number of operators or the number of operators. The experiment is also unclear because the experiment was performed on a different machine than the one used to evaluate the results presented in the paper. The results are not convincing, as performance on the machine was limited by the number of operators used, not by the number of regular expressions used. The paper is not clear as to whether it is possible to use a subset of operators in the same regular expression, thus, there are no guarantees that performance will be comparable to that of a single operator. The experiment is also not convincing because it is unclear whether the proposed approach will work on other kinds of regular expressions. The results are not convincing, because performance on the machine was limited by the number of operators used, not by the number of regular expressions used. The paper is not clear as to whether it is possible to use a subset of operators in the same regular expression, thus, there are no guarantees that performance will be comparable to that of a single operator.

Table 1. Fake reviews for [12], recommending acceptance (up) and rejection (down). These correspond to 1-fake-accept and 1-fake-reject, respectively, in Table 1

of the reviews were fake. Users accessed the form anonymously and were asked to self-assess their research experience and their knowledge on the topics of the papers.

We obtained 16 responses from 12 “Experienced researchers”, 3 “Junior researchers”, and 1 “Student (undergrad/PhD)”, corresponding to 75.5 %, 18.8 %, 6.2 % respectively. The self-assessed familiarity with the research topic of the papers was 2 “High”, 6 “Medium”, and 8 “Low”, 12.5 %, 37.5 %, 50.0 % respectively. The corresponding results are in Figure 1.

In order to interpret these results one should determine the minimum acceptable rating for a fake review. In this respect we observe that the real review of paper 1 was rated “Useless” by a significant fraction of users. If we consider this rating as the minimum acceptable one, then the experiment has been highly successful: only a small percentage of answers rated fake reviews as “Completetely Useless”. Even if we consider the much more challenging baseline corresponding to at least “Useful”, though, it seems fair to claim that the experiment has proven the potential relevance of fake reviews: approximately 75 %, 30 %, and 25 % of the answers for the three papers, respectively, has satisfied this baseline. Interestingly, for paper 1, the fake reviews were deemed more useful overall than the real one.

Some of the optional comments provided by users were quite interesting. Only one of them (experienced researcher, low familiarity) mentioned the possibility that some reviews might have been generated automatically, yet the corresponding ratings for fake

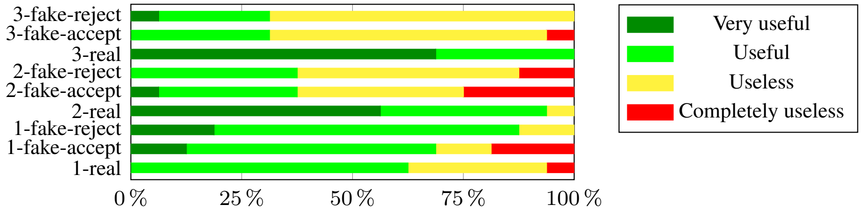

Figure 1. Summary of user study (see the text for a description)

reviews were 1 “Very Useful”, 3 “Useful”, 2 “Useless” without any “Completely Useless” rating. Another user (experienced researcher, medium familiarity) commented that although he/she provided 3 “Useless” ratings, all reviews should be rated at least “Useful” because “they all provide at least some reasons (even vague ones) to accept/reject and specific elements to clarify/explain; real reviews often lack that very basics.”. It is unclear whether the mention of “real reviews” implies that this user realized that some reviews were generated automatically, but this comment is quite interesting anyway.

3. Discussion and concluding remarks

Modern tools for natural language processing and natural language generation may enable novel forms of scholarly fraud based on the automatic generation of fake review reports for academic papers. Although our small user study cannot certainly be conclusive, we have shown that a simple heuristic based on widely available and easy to use tools may be remarkably effective. Such heuristics may be improved in a variety of ways and, most importantly, the power of the language generation engine may be boosted by a new language model much more powerful than GPT-2 tha has been recently announced [15]. This new model, called GPT-3, consists of 175 billion parameters in its larger version—two orders of magnitude bigger than GPT-2. GPT-3 will be made available as a paid cloud-based service and is currently available to a small set of researchers in a wait list.

Our analysis thus indicates that academic frauds based on fake reviews could indeed be feasible and ready to be deployed in the wild. We suggest that journal publishers and conference organizing committees could occasionally inject fake reviews in the reviewing process, in a carefully controlled way, to make sure that these reviews are indeed spotted and discarded. We believe that procedures of this kind could consolidate the quality of a publishing venue and be useful for the scientific community as a whole.

References

[1] Charles W Fox. Difficulty of recruiting reviewers predicts review scores and editorial decisions at six journals of ecology and evolution. *Scientometrics*, 113(1):465–477, 2017.

[2] Charles W Fox, Arianne YK Albert, and Timothy H Vines. Recruitment of reviewers is becoming harder at some journals: A test of the influence of reviewer fatigue at six journals in ecology and evolution. *Research Integrity and Peer Review*, 2(1):3, 2017.

[3] Sameer B Raniga. Decline to Review a Manuscript: Insight and Implications for AJR Reviewers, Authors, and Editorial Staff. *American Journal of Roentgenology*, 214(4):723–726, 2020.

[4] Alberto Bartoli, Andrea De Lorenzo, Eric Medvet, and Fabiano Tarlao. Your paper has been accepted, rejected, or whatever: Automatic generation of scientific paper reviews. In *Availability, Reliability, and Security in Information Systems*, pages 19–28, Cham, 2016. Springer International Publishing.

[5] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. 2019.

[6] Alec Radford. Better language models and their implications. https://openai.com/blog/better-language-models/, February 2019. Accessed: 2020-7-8.

[7] Dongyeop Kang, Waleed Ammar, Bhavana Dalvi, Madeleine van Zuylen, Sebastian Kohlmeier, Eduard Hovy, and Roy Schwartz. A Dataset of Peer Reviews (PeerRead): Collection, Insights and NLP Applications. In *Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, New Orleans, USA, June 2018.

[8] Max Woolf. How to make custom AI-Generated text with GPT-2. https://minimaxir.com/2019/09/howto-gpt2/, September 2019. Accessed: 2020-7-8.

[9] Ondřej Dušek, Jekaterina Novikova, and Verena Rieser. Evaluating the state-of-the-art of End-to-End natural language generation: The E2E NLG challenge. *Comput. Speech Lang.*, 59:123–156, January 2020.

[10] Dimitra Gkatzia and Saad Mahamood. A snapshot of NLG evaluation practices 2005–2014. In *Proceedings of the 15th European Workshop on Natural Language Generation (ENLG)*, pages 57–60, 2015.

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

[12] Alberto Bartoli, Andrea De Lorenzo, Eric Medvet, and Fabiano Tarlao. Playing regex golf with genetic programming. In *Proceedings of the 2014 conference on Genetic and evolutionary computation*, pages 1063–1070. ACM, 2014.

[13] Alberto Bartoli, Andrea De Lorenzo, Eric Medvet, and Fabiano Tarlao. Syntactical similarity learning by means of grammatical evolution. In *International Conference on Parallel Problem Solving from Nature*, pages 260–269. Springer, 2016.

[14] Alberto Bartoli, Andrea De Lorenzo, Eric Medvet, and Fabiano Tarlao. Active learning of regular expressions for entity extraction. *IEEE transactions on cybernetics*, 48(3):1067–1080, 2017.

[15] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.