Thomson, Craig, Ehud Reiter, and Anya Belz. 2024. Common Flaws in Running Human Evaluation Experiments in NLP. *Computational Linguistics*, uncorrected proof.

Common Flaws in Running Human Evaluation Experiments in NLP

Craig Thomson
Department of Computing Science
University of Aberdeen
c.thomson.nlp@gmail.com

Ehud Reiter
Department of Computing Science
University of Aberdeen
e.reiter@abdn.ac.uk

Anya Belz
ADAPT, Dublin City University and
University of Aberdeen
anya.belz@adaptcentre.ie

While conducting a coordinated set of repeat runs of human evaluation experiments in NLP, we discovered flaws in every single experiment we selected for inclusion via a systematic process. In this squib, we describe the types of flaws we discovered, which include coding errors (e.g., loading the wrong system outputs to evaluate), failure to follow standard scientific practice (e.g., ad hoc exclusion of participants and responses), and mistakes in reported numerical results (e.g., reported numbers not matching experimental data). If these problems are widespread, it would have worrying implications for the rigor of NLP evaluation experiments as currently conducted. We discuss what researchers can do to reduce the occurrence of such flaws, including pre-registration, better code development practices, increased testing and piloting, and post-publication addressing of errors.

1. Introduction

Natural language processing (NLP) as a field places great emphasis on human evaluation experiments. In order to be meaningful, experiments must be both designed well and carried out well. Experimental design issues such as the choice of evaluation metric (e.g., Kocmi et al. 2021) and human evaluation method (e.g., Freitag et al. 2021) are often discussed in the NLP literature, as is, more recently, the question of how these impact the reproducibility of results (e.g., Belz et al. 2021; Gundersen et al. 2023). However, there has been little previous discussion of flaws in the running, or execution,

Action Editor: Submission received: 15 June 2023; revised version received: 27 October 2023; accepted for publication: 25 November 2023. **Q1**

https://doi.org/10.1162/coli_a_00508

of experiments, where an experiment is flawed not because the fundamental design is wrong, but because it is poorly implemented or carried out. Part of the reason for this is that reported results are normally taken at face value and the details of how they were produced (such as code used for preparing and processing experimental data) are rarely scrutinized, even if they are made available.

As part of the ReproHum Project[1] on the reproducibility of human evaluations in NLP, we are currently coordinating the efforts of 20 partner NLP labs in a multi-lab multi-test (MLMT) study of the repeatability and reproducibility of human evaluation results reported in NLP venues. For Phase 1 of the study, we selected original evaluation experiments from recent papers published in *ACL* conference proceedings and in the *Transactions of the ACL (TACL)* journal, following a systematic search and selection procedure, as described in detail in Belz et al. (2023). Given that the experiments were selected systematically, the flaws we report here are not from a cherry-picked set of papers, which makes it all the more surprising that we found some form of flaw in every single experiment, ranging from coding errors to reporting errors. Other reproduction studies have also found similar flaws in individual experiments (see Section 2).

In this squib, we report a representative sample of flaws of the different types we discovered in the ReproHum MLMT Study papers, alongside examinations of how they occurred and how they might have been avoided (Section 3). We conclude with a set of recommendations for how to reduce the occurrence of such flaws (Section 4).

Terminology. We use the terms rerunning, repeating, repeatability, and reproducibility in accordance with the International Vocabulary of Metrology (VIM) with the following meanings (following Belz 2022). **Rerunning**, or **repeating**, an experiment means running a given experiment again in exactly the same way except for clearly stated differences that occurred not by design, but due to unavoidable circumstances, such as the same evaluators not being available, a computational tool no longer being accessible, and so forth. Repeatability and reproducibility are properties of measurements, in the present context evaluations that produce numerical values. **Measurement repeatability** (or **repeatability**, for short) is measurement precision under a set of repeatability conditions of measurement, that is, in our case when an evaluation is rerun (2.21 in VIM). **Measurement reproducibility** (**reproducibility** for short) is measurement precision under reproducibility conditions of measurement (2.25 in VIM). Such conditions typically include differences between original and reproduction experiments introduced by design.

Note on Anonymity. In this squib, we deliberately do *not* identify the papers in which we found flaws. One important reason is that such flaws perhaps occur in every researcher's work at some point, and naming a small number of individual authors may give the opposite impression. There is a lack of evidence to suggest that NLP experimental results in NLP experiments *are* reliable, with the evidence presented here showing that in the papers where we thoroughly checked for flaws (by repeating experiments), we found flaws in all papers. We checked with the authors and 3 were happy to be cited, 2 were not, and 1 was indifferent. Therefore, we decided not to cite any authors lest individual authors be identified by exclusion. Also, as described in Belz et al. (2023), only 13% of the authors we contacted were willing and able to give us the information we needed to repeat their experiments. It is unreasonable to assume that experiments

1 https://reprohum.github.io/.

where the resources have not been made available have fewer flaws than those where the resources are available. We are grateful to these authors for their help and support, could not have done this research without their help, and are concerned that criticizing their research by name is unfair and may make authors even less willing to collaborate with future repetition or reproduction of experiments.

2. Related Work

Flaws in the running of experiments occur throughout science. Prominent examples of such flaws include the apparent detection of faster-than-light neutrinos,[2] which was due to a wrongly attached cable and misbehaving clock oscillator; and a flawed economic analysis (which encouraged austerity policies) caused by several rows of data being ignored during analysis.[3] In a medical context, Ioannidis (2005) and Pfeiffer and Hoffmann (2009) argue that reliability of findings published in the scientific literature decreases with the popularity of a research field, in part because competition leads to corner-cutting and even cheating, and in part because if many people do the same type of experiment, this increases the chances (from a statistical perspective) of getting an experiment with misleading results. Carlisle (2021) identified flaws in 44% of medical trials submitted to the Journal *Anaesthesia* between February 2017 to March 2020, where individual patient data was made available; this is compared to 2% when it was not. In other words, the more data they had within which to look for flaws, the more flaws they found. Oransky (2022) believes that 2% of scientific papers should be retracted (i.e., they are so flawed that they need to be withdrawn rather than revised).

In NLP, we are not aware of previous work specifically on flaws in running experiments. However, reproduction studies regularly find software bugs in NLP systems (Arvan, Pina, and Parde 2022b; Papi et al. 2023), which is worrying; see also Raff and Farris (2023). Flaws are occasionally reported in errata to published papers such as Warstadt et al. (2021), but published errata are unusual in the NLP literature. Indeed, Warstadt et al. (2021) is the *only* errata for a flaw in running the experiment (in this case a reporting error) which appeared in *TACL* from 2013 to 2022.[4]

Much more has been published on experimental *design* issues (Howcroft et al. 2020; Shimorina and Belz 2022; Gehrmann, Clark, and Sellam 2023). Sometimes the distinction between design and execution is fuzzy. If there is no pre-registration, then it can be difficult to determine what a researcher intended to do (by design), and therefore whether what they actually did when conducting the experiment differed from the design therefore resulting in an (unintentional) flaw.[5]

3. Flaws Encountered

In this section we describe the types of flaws we found in the six experiments we selected for Phase 1 of the ReproHum MLMT study following a systematic search and selection process (Belz et al. 2023). All experiments were from recent (2018–2022) papers

2 https://en.wikipedia.org/wiki/Faster-than-light_neutrino_anomaly.
3 https://www.bbc.co.uk/news/magazine-22223190.
4 Personal communication from Cindy Robinson, *TACL* editorial assistant. Incidentally, the *ACL Anthology* just includes the uncorrected version of the paper, with no erratum or updated version.
5 As an example, we saw an experiment where items were not shuffled among participants (despite this being good practice), and there being evidence in the resources shared with us that a shuffle was at least considered.

Table 1
Number of flaws of each type per anonymised experiment. Paper F was not selected for Phase 1, but is included as it makes for a good example (see MAD exclusion in Section 3.3).

Paper	Experiment	ReproHum Phase 1	Flaw Type(s)
A	1	Yes	Response collection flaw, Inappropriate exclusion
B	2	Yes	Reporting flaw
C	3	Yes	Coding error
D	4	Yes	Reporting errors (3), Ethical flaw
E	5	Yes	Inappropriate exclusion
E	6	Yes	Coding errors (2), Inappropriate exclusion
F	7	No	Inappropriate exclusion

```
Game A text: The L.A. Lakers defeated the Miami Heat 109 - 106 on Friday .
Anthony Davis recorded 19 points and 23 rebounds .
Game B text: The Utah Jazz defeated the Orlando Magic 123 - 113 on Monday .
```

Figure 1
Example texts (partial basketball summaries) used to illustrate flaws in running experiments.

in ACL conference proceedings or the *TACL* journal. Table 1 shows which types of flaws were seen in these experiments, as well as in an additional experiment (F) that we include here because it serves as a useful example for data point exclusion in Section 3.3.

In keeping with our principle of anonymity (Section 1), most examples presented are situated in the domain of evaluating textual summaries of basketball games, more specifically in the partial summaries shown in Figure 1.

3.1 Errors in Code

We know that researchers make coding mistakes in other contexts (Arvan, Pina, and Parde 2022b), and software engineering tells us to expect around 1–2 defects per 100 lines of code in software that has not gone through a rigorous quality assurance process (McConnell 2004). Very few research systems go through any kind of code review, l et alone commercial-grade software testing and quality assurance, so it seems possible that many experiments may suffer from coding errors.

Figure 2 shows a coding error (in anonymized and simplified form) that was found in one of the experiments we have already repeated. Essentially, the code builds a Python nested dictionary from a CSV file where each line represents a factual error found in a text by a human annotator, including token position data (example shown in Table 2). The bug is that the code builds a two-level nested dictionary (game, sentence), with the effect that if a sentence contains more than one error, only the type of the last error is recorded in the dictionary. This results in incorrect findings when the dictionary is used to count errors and otherwise analyse the data. The code should instead (other coding options exist) build a three-level dictionary, with the third level capable of recording multiple error types (e.g., via a unique identifier for each error in a sentence).

We also found a coding error in a command line script in the GitHub repository for a paper, which the authors informed us was not present when the actual experiment was run, but was introduced when they documented their code after their experiment. Errors in command-line arguments should be considered coding errors because they can (and should) be recorded in shell scripts for experimental repeatability.

```
from csv import DictReader
from collections import defaultdict
error_dict = defaultdict(dict)
# ***Incorrect code. Should be
# error_dict = defaultdict(lambda: defaultdict(dict))
with open("basketball.csv‘‘, ’’r", encoding="utf-8") as fh:
    for r in DictReader(fh):
        error_dict[r[‘‘game’’]][r[‘‘sent’’]] = r[‘‘type’’]
        # ***Incorrect code. Should be
        # error_dict[r[‘‘game’’]][r[‘‘sent’’]][r[‘‘tok_start’’]] = r[‘‘type’’]
```

Figure 2
Buggy code for recording error annotations. This code goes through the data in Table 2, building a 2-level dictionary (game, sentence) from the annotation data. This is used for further analysis, including counting errors. However, this fails when a sentence contains more than one error; in such cases only the last error is recorded. The code should use a 3-level dictionary (game, sentence, tok_start). Corrected lines are included as code comments.

Table 2
Table showing error annotations produced by a human annotator on the texts in Figure 1; each annotation shows a factual error in the text as per Thomson, Reiter, and Sundararajan (2023). This file is stored as `"basketball.csv"`.

game	sent	tok_start	tok_end	error_text	correction	type
A	1	10	10	106	98	NUMBER
A	1	12	12	Friday	Thursday	NAME
A	2	20	20	23	21	NUMBER
B	1	6	7	Orlando Magic	Boston Celtics	NAME

An anonymized version of the flaw is shown in Figure 3. Essentially, this experiment asks Amazon Mechanical Turk workers to compare the outputs of four different NLG systems (`baseline`, `sys1`, `sys2`, `sys3`) as well as human-written `gold` texts. The outputs of all of these systems are computed and stored in corresponding `txt` files, and `prepareMTurkFiles.py` is then used to create a Mechanical Turk experiment from these files. Because of the mistake in line 6 (`-sys3 data/sys2.txt`), this script produces an experiment where Turkers are shown outputs from `sys2` twice, and are never shown outputs from `sys3`.

```
python prepareMTurkFiles.py \
-gold data/gold.txt \
-baseline data/baseline.txt \
-sys1 data/sys1.txt \
-sys2 data/sys2.txt \
-sys3 data/sys2.txt \ #Incorrect code; sys2.txt should be sys3.txt
-output_file outputs/mturk.csv
```

Figure 3
Item combinations created for an experiment by a flawed script. `-sys3 data/sys2.txt` should be `-sys3 data/sys3.txt`.

Game	Fluency (0-2)	Coherence (0-2)	Informativeness (0-2)
A	2,2	2,2	1,2
B	2	2	1

Figure 4
Bad interface design: The participant is asked to determine the number of sentences in Figure 1 and then input a comma-separated list of integer scores for Fluency, Coherence, and Informativeness of each sentence in the text. The first number in the list is supposed to be for the first sentence, the second number is supposed to be for the second sentence, etc.

3.2 Flaws in Response Collection

Responses collected from evaluators can end up flawed if the user interface (UI) is confusing, or makes it hard to enter ratings correctly. Figure 4 shows an anonymized version of a rating interface from one of our experiments. The participant is asked to rate texts in terms of Fluency, Coherence, and Informativeness by entering sentence-level judgments as a comma-separated list, in one cell per quality criterion and text. For example, if the text has two sentences, the participant is expected to enter a comma-separated list with two values for each quality criterion. This is an error-prone way to enter responses; it would be safer to enter scores separately for each sentence (e.g., from a pull-down menu).

This flaw was detected when participants were observed entering incorrect ratings via the UI, for example, recording only two scores when there were three sentences. If individuals struggle to use the experimental UI and/or rating instrument, then collected responses are likely to contain errors such as misaligned or missing evaluation scores. It certainly makes the evaluator's task cognitively harder as they need to keep track of sentence numbers and orders as well as rating them. A flawed UI is a design error more than an execution flaw, but we include this error type, because it can lead to errors during execution when evaluators struggle to use the UI.

3.3 Inappropriate Exclusion of Evaluators and/or Data Points

Researchers do sometimes need to exclude evaluators if they have not taken the experimental task seriously, for example, by randomly clicking on ratings instead of actually making judgments, or if they make too many data entry mistakes. However, exclusion policies need to be established before the experiment is run and should follow best-practice methods such as using predefined attention checks, or using the IQR[6] rule to identify outliers. Otherwise, there is a danger that data and results may be biased.

Table 3 shows an anonymized example of an inappropriate exclusion process. Here, the researchers collected Fluency judgments from four annotators for each item, and dropped as outliers all data points exceeding one median absolute deviation (MAD). This is an inappropriate outlier policy that excludes far too many points. Standard practice for MAD-based exclusion is to treat as outliers points that are >3*MAD from the median (Leys et al. 2013); in the present example, no data points would be excluded from Table 3 using this rule. Indeed, arguably it does not make sense to try to identify

6 https://en.wikipedia.org/wiki/Interquartile_range#Outliers.

Table 3
Per-sentence annotations [0-2] for *fluency* for sentences in Figure 1 by four annotators (*a1*, *a2*, *a3*, and *a4*). Intra-item (sentence) judgments are treated as outliers and excluded (struck through in table) if they are more than one 1 median average deviation (MAD) from the median. This exclusion leads to far higher inter-annotator agreement as measured by Krippendorff's alpha.

game	sent	a1	a2	a3	a4	median	median avg dev	1-MAD range	3-MAD range
A	1	2	2	~~0~~	1	1.5	0.75	0.75 to 2.25	−0.75 to 3.75
A	2	2	2	0	0	1	1.000	0 to 2	−2 to 4
B	1	1	1	~~2~~	~~0~~	1	0.5	0.5 to 1.5	−0.5 to 2.5

outliers at all in a set of four numbers. In the present example, >1*MAD outlier exclusion resulted in much higher inter-annotator agreement than would have been the case if >3*MAD had been used. We detected this error when trying to recreate the figures in the paper from the raw annotation data; when our results did not match those reported in the paper, we investigated and discovered the above outlier issue.

Other non-standard practices in the exclusion of participants found in ReproHum papers included ad-hoc attention checks where participants were excluded if researchers thought their answers were too far from expected values, with no clear procedure recorded to explain which answers should trigger exclusion. In another case, only the annotations by the participant who was considered the most experienced in the researchers' opinion were retained in their entirety when computing final results.

3.4 Reporting Flaws

Another type of flaw we found was incorrect reporting of results in publications, where numbers reported in the paper do not match the numbers present in, or derivable from, the experimental data and other resources. For example, one paper stated that 50 input items were selected from the dataset for each of two modes when it was actually 50 total input items (25 per mode). In another paper, the performance of a system (a percentage) differed between the text and a figure. In a third case, the authors themselves found the reporting flaw prompted by us contacting them about repeating their experiment.

3.5 Ethical Flaws

Some flaws relate to research ethics instead of the validity of experimental results, for example, when non-anonymized data is made publicly available. This was the case in one paper where the worker IDs for Amazon Mechanical Turk workers were included. These 14-character unique worker identifiers are *not* anonymous (Lease et al. 2013) and should always be anonymized before data is published. Indeed, if a project has been approved by a research ethics board, then failure to anonymize WorkerID may violate the conditions of such approval.

4. Reducing the Likelihood of Flaws when Running Experiments

With hindsight, it is straightforward to see fixes for the flaws discussed in Section 3. However, it is more challenging to get everything right ahead of time, and there are unfortunately no "magic bullets" that can prevent all possible flaws in future NLP

experiments. Below we list some recommendations for what can be done at different points in the development and running of a human evaluation experiment to prevent some common flaws including those described in Section 3.

i. Check Evaluation Items. When preparing evaluation items (system outputs, usually), simple tests can be written to confirm items exhibit the desired properties. In relative evaluations, for example, such tests could ensure that outputs being compared are distinct; this would have caught the code flaw discussed at the end of Section 3.1.

ii. Use Code Development Best Practices. Most researchers are not professional software developers, but we still need to use good software engineering practices. While academic coders cannot normally spend as much time testing and debugging code as their commercial counterparts, basic good coding practices need not take a huge amount of time. At a minimum, every piece of code should be seen by another researcher either reviewing the code or writing a second version of it to compare results. Mineault and Community (2021) provide a good resource aimed at researchers.

iii. Use Safe UIs for Response Collection. Participants should not be able to introduce errors in their responses. User testing (e.g., during a pilot) can identify ways in which participants might accidentally enter invalid responses as seen in Section 3.2. Automated form validation can also be used. However, it is safest to use mechanisms such as menus of rating options that ensure that all responses entered are valid.

iv. Run a Pilot. A pilot version of the experiment should always be run first, and results from it checked by both manual and automatic methods (Arvan, Pina, and Parde 2022a). A pilot almost inevitably results in improvements in the experiment and can reveal different types of flaws, such as the code flaws described in Section 3.1.

v. Use Pre-registration and Avoid ad-hoc Changes. Ad-hoc decisions made as issues arise during experiment development and execution are likely to lead to experimental flaws; these could be avoided if the experimental process is recorded in advance, for example with a pre-registration (van Miltenburg, van der Lee, and Krahmer 2021), and then followed exactly by researchers. The very process of sitting down and documenting these steps, ideally with a discussion between authors, should help identify possible flaws in the process, such as inappropriate outlier exclusion or attention check policies, as discussed in Section 3.3.

vi. Double-check Reported Results and Experimental Resources. Human error is hard to avoid when manually transcribing values. Reporting errors (Section 3.4) can be reduced by using automatic processes for creating tables and figures. The results processing script can directly output tables, for example, in LaTex format using tools such as *knitr*, *Sweave*, *PyLaTeX*, and *pandas.DataFrame.to_latex*. Having multiple authors check data and resource files can also reduce the likelihood of ethical flaws such as the failure to anonymize data discussed in Section 3.5.

vii. Share All Experimental Resources Including Raw Responses. Carlisle (2021) found around 20 times as many flaws in trials where raw patient data (roughly equivalent to raw evaluator responses in NLP) was available; we suggest that authors make annotation data, and any code used to process it, available by default in anonymized

form. This is especially important since our experience (Belz et al. 2023) has been that authors struggle to produce such information when contacted after the paper has been published. This would make it easier for other researchers to verify results.

viii. Engage in Post-publication Review and Discussion. NLP as a field does not have mechanisms for discovering and addressing flaws and errors in papers after publication. Good medical journals associate discussion forums with papers, where readers can ask questions and raise concerns about papers *after* they are published. Authors are expected to take these seriously and respond. NLP conferences and journals do not currently support this; ICLR allows post-publication discussion using OpenReview, although this is not monitored after publication. At a minimum, NLP would benefit from standard mechanisms for reporting and correcting errors post-publication.

5. Conclusion

Experimental rigor is essential in science, and this applies to NLP as it does to fields like medicine and physics. We have discovered flaws in the running of some experiments in every paper we selected for the ReproHum MLMT study, mostly of a type that cannot be detected by current reviewing practices. We hope that our discussion of these flaws will help researchers both do more rigorous experiments themselves and make them aware of potential problems in papers by other authors.

Acknowledgments

The ReproHum project is funded by EPSRC grant EP/V05645X/1. We would first like to thank all authors who took the time to respond to our requests for information; we could not have done this work without their help! We would also like to thank all the people at the ReproHum partner labs who helped us carry out Phase 1 of the MLMT study; Gavin Abercrombie, Jose M. Alonso-Moral, Mohammad Arvan, Anouck Braggaar, Jackie Cheung, Mark Cieliebak, Elizabeth Clark, Kees van Deemter, Tanvi Dinkar, Ondřej Dušek, Steffen Eger, Qixiang Fang, Mingqi Gao, Albert Gatt, Dimitra Gkatzia, Javier González-Corbelle, Dirk Hovy, Manuela Hürlimann, Takumi Ito, John D. Kelleher, Emiel Khramer, Filip Klubicka, Huiyuan Lai, Chris van der Lee, Yiru Li, Saad Mahamood, Margot Mieskes, Emiel van Miltenburg, Pablo Mosteiro, Malvina Nissim, Natalie Parde, Ondřej Plátek, Verena Rieser, Jie Ruan, Joel Tetreault, Antonio Toral, Xiaojun Wan, Leo Wanner, Lewis Watson, and Diyi Yang. Special thanks to Mohammad Arvan, Saad Mahamood, Emiel van Miltenburg, Natalie Parde, Barkavi Sundararajan, as well as the action editor and anonymous reviewers for their very helpful suggestions for improving this paper. We also thank Cindy Robinson for providing information about errata in *TACL*.

References

Arvan, Mohammad, Luís Pina, and Natalie Parde. 2022a. Reproducibility in computational linguistics: Is source code enough? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2350–2361. https://doi.org/10.18653/v1/2022.emnlp-main.150

Arvan, Mohammad, Luís Pina, and Natalie Parde. 2022b. Reproducibility of exploring neural text simplification models: A review. In *Proceedings of the 15th International Conference on Natural Language Generation: Generation Challenges*, pages 62–70.

Belz, Anya. 2022. A metrological perspective on reproducibility in NLP. *Computational Linguistics*, 48(4):1125–1135. https://doi.org/10.1162/coli_a_00448

Belz, Anya, Shubham Agarwal, Anastasia Shimorina, and Ehud Reiter. 2021. A systematic review of reproducibility research in natural language processing. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 381–393. https://doi.org/10.18653/v1/2021.eacl-main.29

Belz, Anya, Craig Thomson, Ehud Reiter, Gavin Abercrombie, Jose M. Alonso-Moral, Mohammad Arvan,

Anouck Braggaar, Mark Cieliebak, Elizabeth Clark, Kees van Deemter, Tanvi Dinkar, Ondrej Dušek, Steffen Eger, Qixiang Fang, Mingqi Gao, Albert Gatt, Dimitra Gkatzia, Javier González-Corbelle, Dirk Hovy, Manuela Hürlimann, Takumi Ito, John D. Kelleher, Filip Klubicka, Emiel Krahmer, Huiyuan Lai, Chris van der Lee, Yiru Li, Saad Mahamood, Margot Mieskes, Emiel van Miltenburg, Pablo Mosteiro, Malvina Nissim, Natalie Parde, Ondrej Plátek, Verena Rieser, Jie Ruan, Joel Tetreault, Antonio Toral, Xiaojun Wan, Leo Wanner, Lewis Watson, and Diyi Yang. 2023. Missing information, unresponsive authors, experimental flaws: The impossibility of assessing the reproducibility of previous human evaluations in NLP. In *The Fourth Workshop on Insights from Negative Results in NLP*, pages 1–10. https://doi.org/10.18653/v1/2023.insights-1.1

Carlisle, J. B. 2021. False individual patient data and zombie randomised controlled trials submitted to anaesthesia. *Anaesthesia*, 76(4):472–479. https://doi.org/10.1111/anae.15263, PubMed: 33040331

Freitag, Markus, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474. https://doi.org/10.1162/tacl_a_00437

Gehrmann, Sebastian, Elizabeth Clark, and Thibault Sellam. 2023. Repairing the cracked foundation: A survey of obstacles in evaluation practices for generated text. *Journal of Artificial Intelligence Research*, 77:103–166. https://doi.org/10.1613/jair.1.13715

Gundersen, Odd Erik, Kevin Coakley, Christine Kirkpatrick, and Yolanda Gil. 2023. Sources of irreproducibility in machine learning: A review. Available online at https://arxiv.org/abs/2204.07610v2.

Howcroft, David M., Anya Belz, Miruna-Adriana Clinciu, Dimitra Gkatzia, Sadid A. Hasan, Saad Mahamood, Simon Mille, Emiel van Miltenburg, Sashank Santhanam, and Verena Rieser. 2020. Twenty years of confusion in human evaluation: NLG needs evaluation sheets and standardised definitions. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 169–182. https://doi.org/10.18653/v1/2020.inlg-1.23

Ioannidis, John P. A. 2005. Why most published research findings are false. *PLoS Medicine*, 2(8):e124. https://doi.org/10.1371/journal.pmed.0020124, PubMed: 16060722

Kocmi, Tom, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494.

Lease, Matthew, Jessica R. Hullman, Jeffrey P. Bigham, Michael S. Bernstein, Juho Kim, Walter S. Lasecki, Saeideh Bakhshi, Tanushree Mitra, and Robert C. Miller. 2013. Mechanical Turk is not anonymous. *Entrepreneurship & Economics eJournal*. https://doi.org/10.2139/ssrn.2228728

Leys, Christophe, Christophe Ley, Olivier Klein, Philippe Bernard, and Laurent Licata. 2013. Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. *Journal of Experimental Social Psychology*, 49(4):764–766. https://doi.org/10.1016/j.jesp.2013.03.013

McConnell, Steve. 2004. *Code Complete*, 2nd ed. Microsoft Press.

Mineault, Patrick J. and The Good Research Code Handbook Community. 2021. The good research code handbook. Zenodo. Available at https://goodresearch.dev.

Oransky, Ivan. 2022. Retractions are increasing, but not enough. *Nature*, 608(7921):9. https://doi.org/10.1038/d41586-022-02071-6, PubMed: 35918520

Papi, Sara, Marco Gaido, Andrea Pilzer, and Matteo Negri. 2023. When good and reproducible results are a giant with feet of clay: The importance of software quality in NLP. Preprint available online at https://arxiv.org/abs/2303.16166.

Pfeiffer, Thomas and Robert Hoffmann. 2009. Large-scale assessment of the effect of popularity on the reliability of research. *PLoS One*, 4(6):e5996. https://doi.org/10.1371/journal.pone.0005996, PubMed: 19551148

Raff, Edward and Andrew L. Farris. 2023. A siren song of open source reproducibility, examples from machine learning. In *Proceedings of the 2023 ACM Conference on Reproducibility and Replicability*, ACM REP '23, pages 115–120. https://doi.org/10.1145/3589806.3600042

Shimorina, Anastasia and Anya Belz. 2022. The human evaluation datasheet: A template for recording details of human evaluation experiments in NLP. In *Proceedings of the 2nd Workshop on Human Evaluation of NLP Systems (HumEval)*, pages 54–75. https://doi.org/10.18653/v1/2022.humeval-1.6

Thomson, Craig, Ehud Reiter, and Barkavi Sundararajan. 2023. Evaluating factual accuracy in complex data-to-text. *Computer Speech & Language*, 80:101482. https://doi.org/10.1016/j.csl.2023.101482

van Miltenburg, Emiel, Chris van der Lee, and Emiel Krahmer. 2021. Preregistering NLP research. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 613–623. https://doi.org/10.18653/v1/2021.naacl-main.51

Warstadt, Alex, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2021. Erratum: BLiMP: The Benchmark of Linguistic Minimal Pairs for English. *Transactions of the Association for Computational Linguistics*, 8:867–868. https://doi.org/10.1162/tacl_x_00375