From Detection to Correction: A Hybrid NLP Approach to Misinterpretations of Nonsignificant *p* Values

Raphael Merz

Department of Psychology, Ruhr University Bochum

Author Note

Raphael Merz https://orcid.org/0000-0002-9474-3379

Correspondence concerning this article should be addressed to Raphael Merz, Email: raphael.merz@rub.de

Abstract

Misinterpretations of *p* values remain widespread in scientific reporting, despite decades of educational efforts and reform initiatives. One of the most common and consequential errors is interpreting a statistically nonsignificant result (e.g., $p > .05$) as evidence for the absence of an effect—a conclusion not supported by null hypothesis significance testing (NHST). This thesis adopts a human factors perspective, arguing that automation can help mitigate such persistent errors, much like word processors assist with grammar and spelling. I propose an automated, three-step pipeline that detects, classifies, and optionally corrects misinterpretations of nonsignificant results. Evaluation of each step highlights the promise of such an automated approach: In a validation set of 25 articles the automatic detection identified 73% of human-extracted statements. Two easily resolvable issues in the search pattern were found which, once addressed, would increase this reliability to 93%. For classification, three BERT-based models were trained on 930 hand-labeled statements. All performed well, with SciBERT achieving the highest macro F1 score of .91. Finally, the optional correction step proved effective in a validation set of 80 incorrect and 20 correcct statements: 85 statements were correctly phrased after LLM-based revision. These results demonstrate that automation can effectively address this specific misinterpretation and offer a flexible foundation for tackling similar issues in scientific writing and meta-research.

Keywords: p value, misinterpretation, automation, automated checks, RegEx, LLMs, BERT

From Detection to Correction: A Hybrid NLP Approach to Misinterpretations of Nonsignificant *p* Values

1 Introduction

Over the past decades, numerous articles have addressed common misinterpretations of *p* values in the context of standard null hypothesis significance testing (NHST; Goodman, 2008; Greenland et al., 2016; Schervish, 1996). Some go further, questioning the use of frequentist methods altogether (Edwards et al., 1963; Wagenmakers, 2007), while others propose refinements within the frequentist framework that aim to improve the informativeness of statistical inference (Isager & Fitzgerald, 2025; Lakens et al., 2018). If you are a researcher writing a paper and want to interpret your results correctly, the solution seems simple: read these educational resources and revise your manuscript accordingly. Easy, right? Still, empirical studies consistently show that these misinterpretations remain widespread (Aczel et al., 2018; Hoekstra et al., 2006; Murphy et al., 2025). Why is that? **What makes interpreting *p* values so persistently difficult?** And which practical solutions or promising approaches might help?

In this article, I show how rule-based approaches, combined with natural language processing (NLP), can be used to automatically detect, classify, and correct these misinterpretations. I focus on the misinterpretation of statistically nonsignificant results as the absence of an effect because it arguably has the strongest impact on researchers' conclusions and is the most extensively studied misinterpretation of *p* values (Lakens, 2021). Similarly, my previous work has developed clear criteria for classifying this misinterpretation (Murphy et al., 2025). I demonstrate how this automated approach may help us to finally overcome this misinterpretation.

1.1 Misinterpretations and Criticism of *P* Values

The criticism of *p* values has become a prominent and recurring theme in discussions around scientific reform. From claims that they encourage dichotomous thinking (Amrhein et al., 2019; Hoekstra et al., 2006) to arguments that they offer little informational value (Wagenmakers, 2007), *p* values – and the broader framework of NHST – have been blamed for many of science's replication problems (McShane et al., 2019). On the other hand, many have

also argued that NHST per se is not to blame for these problems, but rather how researchers (mis)use and (mis)interpret this tool (e.g., Greenland, 2019; Lakens, 2021). As a result, many researchers present whole collections of, in their view, common *p* value misinterpretations (see, e.g., Goodman, 2008; Greenland et al., 2016).

In this study I zoom in on one specific misinterpretation: concluding *no effect* based on a statistically nonsignificant finding. Many studies have previously shown that this misinterpretation remains highly prevalent across time and sub-domains of psychology (Aczel et al., 2018; Hoekstra et al., 2006; Murphy et al., 2025). In fact, in a recently published article investigating articles published in 2009, 2015, and 2021 across ten different psychology journals, we estimated the prevalence of this misinterpretation in articles' discussion sections to lie between 76.17% and 84.90% (Murphy et al., 2025). These findings highlight that the situation seems not to have greatly improved despite continuous calls to reflect on statistical interpretations of nonsignificant results (e.g., McShane et al., 2019, **add other source?**) and and increasing advocacy for alternative analytical approaches that enable researchers to make informed claims about effects being practically equivalent to zero (e.g., **?** add this article https://doi.org/10.3389/fpsyg.2014.00781; Lakens et al., 2018).

1.2 Possible Solutions

One frequently suggested solution is to improve researchers' statistical literacy through enhanced education, such as better statistics teaching at the undergraduate and graduate levels (e.g., Lakens, 2021). However, as noted earlier, the persistent prevalence of the misinterpretation examined in this study indicates that calls for improved education alone have not been sufficient to address the issue (Murphy et al., 2025). This is complemented by research showing that many misinterpretations of *p* values are shared among psychology students and teachers (Badenes-Ribera et al., 2016; Haller & Krauss, 2002). Recognizing the limitations of education alone, researchers have also advocated for the use of interval hypotheses tests, like equivalence testing or minimum-effect tests (or the combination: three-sided testing; Isager & Fitzgerald, 2025). These methods allow researchers to test whether an effect is practically relevant and larger than a predefined smallest effect size of interest (SESOI; Lakens et al., 2018). In many contexts, such approaches might be more

closely aligned with the substantive questions researchers aim to answer, namely whether an effect is meaningful in practice.

These proposed solutions also align with the argument made by Lakens (2021) that *p* value misinterpretations represent a human factors problem, requiring practical and easy-to-implement solutions. In other contexts we encounter systems like this frequently, be it automatic braking systems in cars, word processors that flag spelling and grammar mistakes, or email clients that filter out malware and phishing attempts. Analogously, automated checks for statistical misinterpretations offer a highly promising route. This perspective emphasizes that many statistical errors arise not from bad intentions or ignorance, but from cognitive limitations and suboptimal workflows.

In the context of research, similar automated solutions are already gaining traction. For instance, the reference manager Zotero flags references to retracted papers (Stillman, 2019). Statcheck (Nuijten & Epskamp, 2024) automatically detects inconsistencies between reported test statistics and *p* values. Other tools, like GRIM, GRIMMER, and SPRITE, identify impossible values in reported summary statistics (Heathers et al., 2018). And lastly, Regcheck (Cummin & Hussey, 2025) verifies the consistency between manuscripts and their preregistration documents. As AI continues to develop, we can expect these types of automated solutions to become increasingly sophisticated and common.

Following this trend, DeBruine and Lakens (2025) developed Papercheck, an R package, which allows users to run a battery of automated checks on scientific papers. These include statistical checks (e.g., identifying imprecisely reported *p* values) as well as general manuscript quality checks (e.g., verifying links to online repositories or consistency between in-text citations and reference lists). Papercheck can be used both for single articles (e.g., as writing assistance) and for batches of articles (e.g., for meta-scientific studies). Because this framework is actively maintained and continues to evolve, the approach presented in this study was designed to fit within the Papercheck infrastructure.

–> I originally had a short summary paragraph here to, again, repeat the “research question”/aim and to transition to the methods more smoothly, but Daniel said it wasn’t necessary and too much repetition. Let me know if you feel like its missing here! <–

2 Methods

2.1 Statement Detection, Classification and Correction

To provide context for the data used in this study, I first outline the three steps of the proposed pipeline. Statements from scientific articles need to be reliably detected, classified, and finally, if desired, corrected. For each step, I applied specific methods that were best suited to achieve the respective goal.

To detect statements I used rule-based regular expressions (RegEx) and searched articles' results sections to detect these expressions. Effectively, RegEx searchers are advanced Ctrl+F searches, where a user can include rules like optional characters (e.g., 'significant(ly)' would catch both *significant* and *significantly*) and more complex rules (e.g., 'not.{0,20}significant' allows up to 20 characters between *not* and *significant*). Papercheck has a module that detects almost all *p* values (see Section 3.1 for examples currently not detected) based on RegEx searches. Using this module, I created a subset of all *p* values equal to or above .05.[1] I then expanded the extracted nonsignificant *p* values to the full sentence and added +/- one sentence as context in case of extraction errors (incomplete statements).

In the next step, these statements (labeled as correct or incorrect by me; see Section 2.2) were used to train three BERT-based models. BERT (Bidirectional Encoder Representations from Transformers) is a general-purpose language model pre-trained on the BookCorpus and English Wikipedia, making it suitable for a wide range of tasks – but not specifically optimized for scientific language (Devlin et al., 2019). Since its introduction, many researchers have developed domain-specific variants of BERT to enhance its performance on specialized tasks. To test whether such domain adaptation improves performance in this study's classification task, I trained two models in addition to standard BERT: SciBERT was trained on a large corpus of scientific articles from Semantic Scholar, particularly in the biomedical and computer science domains (Beltagy et al., 2019). PubMedBERT is an even more specific pretrained language model, having been trained exclusively on biomedical abstracts and full-text articles from the PubMed database (Gu et al.,

[1] In a final Papercheck module, users will be able to set the alpha level they used themselves, thus allowing other levels than the conventional 5%.

2022). These models were trained and evaluated on their ability to distinguish between correct and incorrect interpretations of nonsignificant results in scientific writing. The models' hyperparameters (e.g., learning rate, batch size) were informed by established defaults in the field (see https://huggingface.co/docs/transformers/en/main_classes/trainer) and relevant tutorials (More, 2025; Talebi, 2024), with further refinements to improve the models' prediction performance.

Lastly, in the application of this framework, statements classified as incorrect by the best-performing BERT model would be sent to an LLM for correction. However, to assess how the LLM handles both genuinely incorrect statements and those misclassified as incorrect automatically, I submitted both correct and incorrect statements coded by me to the LLM in this study (see Section 2.2 for details).

2.2 Validation Process and Performance Metrics

To assess the effectiveness of each automated approach, I compared their outputs to human-coded ground truth and calculated appropriate reliability and performance metrics. The validation process was conducted separately for statement detection, classification, and correction.

2.2.1 Statement Detection

Firstly, to ensure that the statement detection process caught all statements with nonsignificant *p* values in articles' results sections, I manually extracted these statements from 25 (10%; randomly chosen) of the Papercheck sample library's 250 open access article from the journal Psychological Science. These articles were published between 2013 and 2024 (Median = 2021). I then coded whether statements I found were also extracted with the automated RegEx search.

2.2.2 Statement Classification

For the training of the BERT models and to assess their final performance, I labeled all automatically extracted statements that were detected from an article's results section from Papercheck's sample library. This resulted in 960 statements in total. Of these, 419 were classified as containing a correct *p* value interpretation by me, and 353 were classified as incorrect *p* value misinterpretations. The remaining 188 statements were classified as neither

correct nor incorrect because they interpreted the nonsignificant effect as (marginally) significant (83), because the statements were not complete enough to check their correctness (20), because they interpreted model fit indices and not the *p* value (20), because they were falsely flagged as containing a nonsignificant *p* value (e.g., significant *p* values or generic '*p* > .05 indicated by symbol xy' statements from table/figure notes; 19 in total), or due to a combination of these or other reasons (46).

Before actually training a model, the labeled data was split into three parts: a test set (20%) used for the final evaluation of the model, a training set (72%, or 90% of the remaining 80%) that the model used to learn underlying patterns and adjust its parameters, and a validation set (8%, or 10% of the 80%) used to calculate evaluation metrics after each epoch (i.e., one full cycle of the model processing the training data) to prevent overfitting to the training data. The number of correct and incorrect interpretations was balanced in each of these parts (originally there were more correct than incorrect interpretations) to ensure that the model would not overfit to this class-imbalance (and only predict the majority class).

Before training the model, the labeled data was split into three parts: a test set (20%) for final model evaluation, a training set (72%, or 90% of the remaining 80%) for learning and parameter adjustment, and a validation set (8%, or 10% of the 80%) for calculating evaluation metrics after each epoch (i.e., one full cycle of the model processing the training data) and monitoring overfitting. To address the original class imbalance (with more correct than incorrect statements), the number of correct and incorrect interpretations was balanced in each set, ensuring the model would not simply learn to predict the majority class.

During BERT training, I computed the training loss (sum of errors between model predictions and actual labels in the training set) and the validation loss (same for validation set). The best-performing model was selected based on the lowest validation loss to prevent the model from overfitting to the training data. The model would have been trained on a maximum of 16 epochs, but training ended early if the model did not improve, as measured by the validation loss, for two consecutive epochs. Ultimately, the longest number of training epochs was 7. For the final evaluation, I computed the fraction of correctly predicted classes among all predicted cases of a class (precision), the fraction of correctly predicted classes

among all actual cases of a class (recall), and their harmonic mean (F1 score), separately for each class (incorrect and correct). To summarize overall performance across the two classes, I calculated the unweighted average of the two F1 scores (macro-F1 score).

2.2.2 Statement Correction

Lastly, I reviewed 100 statements that were sent to an LLM for correction to evaluate whether the revised statements were correct. Of these, 80 had previously been labeled incorrect and 20 correct, allowing me to examine how the LLM handled false positives from the automated classification. To communicate with the models, I used Papercheck, which relies on the Groq API (available at https://groq.com/). I tested two LLMs - 'llama-3.3-70b-versatile' (created 03-09-2023) and 'openai/gpt-oss-120b' (created 05-08-2025) - and applied two prompts to the full validation dataset of 100 statements. This resulted in three iterations: (1) the initial prompt with the 'llama-3.3-70b-versatile' model, (2) the same prompt with 'openai/gpt-oss-120b', and (3) a refined prompt, developed through preliminary tests on subsets of the 100 statements, with 'openai/gpt-oss-120b'.

Both prompt versions first explained to the LLM that it would receive a statement containing at least one misinterpretation of a nonsignificant finding as the absence of an effect, and instructed it to revise only the part of the statement containing this misinterpretation, leaving the rest unchanged. The refined prompt included additional guidance on phrasing to avoid, based on common errors that persisted in revisions from the initial prompt. Finally, the initial prompt instructed the LLM to respond with "NO CORRECTION POSSIBLE" if it found no nonsignificant *p* value or interpretation of it to account for possible errors during the automatic detection of statements. This instruction was removed in the refined prompt because the LLM overused this option, which blurred the distinction between statement detection, classification, and correction. Both prompts are available on GitHub (see Section 2.3).

2.3 Software

All scripts for this study were written in R (Version 4.5.0; Team, 2025) or Python (Version 3.12.10; Python Software Foundation, 2025).

In R, I used *papercheck* (Version 0.0.0.9049; DeBruine & Lakens, 2025) for accessing the 250 open access articles, preprocess them and for communication with the LLMs, *readxl*

(Version 1.4.5; Wickham & Bryan, 2025) to access Excel files in R, *psych* (Version 2.5.6; William Revelle, 2025) for calculating descriptive statistics, *tidyverse* (Version 2.0.0; Wickham et al., 2019) for data preprocessing and visualization, and *flextable* (Version 0.9.9; Gohel & Skintzos, 2025), *magick* (Version 2.8.7; Jeroen, 2025), *papaja* (Version 0.1.3; Aust & Barth, 2024) and *showtext* (Version 0.9-7; Qiu & for details., 2024) to create APA-formatted tables and figures.

All scripts and data to reproduce and use the trained BERT models (Python), analyse the results and validity checks (R and Python), recreate this manuscript (Quarto Markdown in R Studio with the apaquarto extension available at: https://wjschne.github.io/apaquarto/), as well as the list of Python libraries used to train the BERT models are available in this GitHub repository, together with instructions on how to set it up: LINK.

Due to the project's iterative nature and since no inferential statistical tests were performed, this study was not preregistered. The original project proposal can also be found in the GitHub repository.

3 Results

3.1 Detection Accuracy

By manually reviewing 25 articles from Papercheck's sample library, I identified 179 statements containing a nonsignificant p value. The automated RegEx search fully detected 130 (73 %) of these, and partially detected 6 due to extraction errors, often caused by PDF formatting (page breaks, figures, or footnotes). The search also produced 3 false positives - statements incorrectly labeled as coming from the results section, but actually originating from other sections, or from table or figure notes. Note, however, that most of the 49 (partially) missed statements were due to specific ways of writing (or not writing) the *p* value: 31 were missed because the *p* value was written as 'p_s', and 8 were missed because the authors wrote 'n.s.' instead of the nonsignificant *p* value. Excluding these two types of reporting, the overall agreement between the automated and manual approaches would have been 93 %.

Lastly, the remaining 10 missed statements were due to pdf formatting issues such as figures, tables, footnotes, page breaks, or unusual characters within the statement that

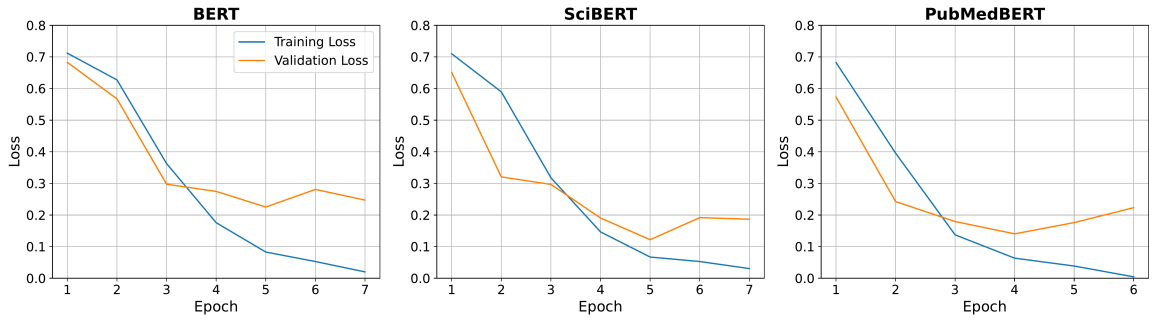
interfered with the statement detection (9 in total).[2]

3.2 Classification Performance

Figure 1 shows the training and validation loss curves for the three BERT models across their training epochs. The standard BERT model was trained for a total of 7 epochs before early stopping was triggered due to a lack of improvement in validation loss for two consecutive epochs. The model from epoch 5 was therefore selected as the best-performing one. Similarly, SciBERT and PubMedBERT reached the lowest validation loss after epochs 5 and 4, respectively. As shown in the figure, the training loss consistently decreased over time for all three models, as expected given that models were optimized to fit the training data. In contrast, the validation loss plateaued in all models before increasing again, indicating that further improvements in fitting the training data no longer translated into better performance on unseen data and may even signal the onset of overfitting.

Figure 1

Training and Validation Loss Curve



Note. Curves of the training and validation loss of the three trained BERT models. The best models for regular BERT, SciBERT and PubMedBERT were chosen after epoch 5, 5, and 4, respectively, based on the minumum validation loss.

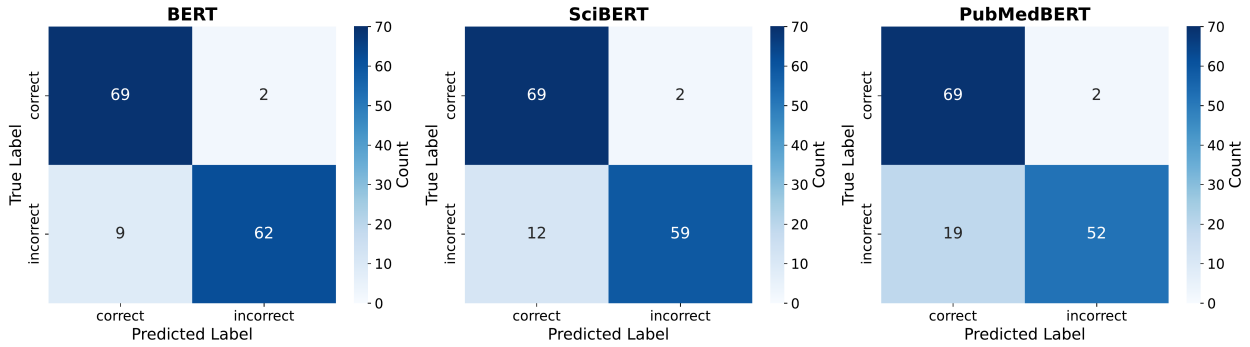
Moving to model performance, Figure 2 shows the number of correctly and incorrectly classified statements for each model. In general, all three models performed well. Overall, the standard BERT model showed the fewest misclassifications with 2 false positives and 9 false

[2] I could not find one statement that was extracted automatically in the article's pdf. My current theory is that this was an artifact from when the pdf was compiled and might be from a different article even, once again highlighting how impractical the pdf format is in times of increasing automation.

negatives, whereas SciBERT and PubMedBERT showed 14 and 21 false classifications in total, respectively. Zooming out, these results are also visible in Table 1, which summarizes the performance metrics. As reflected in the macro F1 score, again, BERT achieved the best overall performance in classifying correct and incorrect statements with a macro F1 score of .92. SciBERT and PubMedBERT lagged slightly behind, with macro F1 scores of .90 and .85, respectively. All three models better predicted correct statements than incorrect ones, reflected by the F1 score of the 'correct' class, with standard BERT scoring best ('correct' F1 score of .93). Similarly, in all models, recall was higher than precision in the 'correct' class, whereas the opposite pattern was visible in the 'incorrect' class, suggesting that the models tend to err on the side of overidentifying statements as correct rather than incorrect. In fact, the standard BERT model was just slightly better at reducing false negatives (at the cost of more false positives) in this test set (Precision in the 'incorrect' class: .97 vs. SciBERT's .97).

Figure 2

Confusion Matrix



Note. Confusion matrices of the three trained BERT models.

Table 2 shows statements misclassified by all three models to illustrate common sources of difficulty. Potential causes of these misclassifications will be explored in the discussion.

3.3 Correction Evaluation

Of the 100 statements that the LLM was instructed to correct 85 were judged as correct. Notably, 2 of the 20 already correct statements were turned incorrect by the LLM, and 13 of the 80 incorrect statements remained incorrect. Using the newer 'openai/gpt-oss-120b' model, 79 were correct. However, the model overused the 'NO CORRECTION POSSIBLE'

Table 1

Model Performance

	BERT			SciBERT			PubMedBERT		
	Precision	Recall	F1 score	Precision	Recall	F1 score	Precision	Recall	F1 score
Correct Class	.88	.97	.93	.85	.97	.91	.78	.97	.87
Incorrect Class	.97	.87	.92	.97	.83	.89	.96	.73	.83
Macro F1 score			.92			.90			.85

Note. Table of precision, recall and F1 score per model and class.

option (originally intended to prevent detection errors) which resulted in 8 correct and 2 incorrect statements being left unrevised. In the final iteration, with the newer model and revised prompt, 93 of the 100 statements were correct. Of the remaining 7 statements, XXX were originally incorrect statement that remained incorrect, whereas the other YYY did not misinterpret nonsignificance but contained other problems: in one case, the LLM removed all *p* values and interpreted the effects as significant, and in another, it altered the statement's meaning substantially. Examples of both poor and strong LLM-revisions from this final iteration are shown in Table 3 and Table 4, respectively.

Above paragraph needs to be updated after discussion on how to code some of them with Daniel! Also need to update below tables after that

4 Discussion

Things I still need do add:

- GO back to human factors component, maybe int he part about LLM-corrections?
- I don't yet discuss why BERT might have show the better performance compared to the other two models
- I don't really discuss the reasons for misclassifications contrary to that I say in the results section

4.1 Summary of Key Results

In this study, I developed and evaluated a three-step pipeline for automatically correcting misinterpretations of nonsignificant results as evidence for the absence of an effect. The approach combines rule-based RegEx searches for detecting candidate statements,

solution for learning subtle language patterns. Finally, the optional LLM correction might enhance the user experience further by offering useful rewording suggestions tailored to the context of the specific statement. This layered, hybrid-NLP structure makes the approach both flexible and easily scalable.

The RegEx-based statement detection phase demonstrated that simple, rule-based searches can effectively flag a large proportion of candidate interpretations. Although formatting issues in pdfs made the correct extraction of these misinterpretations impossible in some cases, the vast majority of statements were automatically detected. In addition, the study revealed straightforward issues in the current approach (e.g., as "p = n.s." or with subscripted 'p_s') that can be fixed with minimal adjustments, further enhancing detection accuracy.

At the same time, the classification results are particularly promising given the relatively small size of the manually labeled dataset (< 1,000 examples, split into training, validation, and test sets). The strong performance likely reflects a certain regularity in how nonsignificance is (mis)interpreted in academic writing - commonly through the use of either 'significant' or 'no effect' (e.g., 'there was no effect', but also 'groups did not differ') terminology. While the training dataset was limited to statements extracted from Psychological Science articles (using the existing Papercheck sample library), the results provide a solid baseline for expansion using more diverse sources and research domains.

The final step, generating LLM-revised corrections of the original statements, showed clear promise, indicating that LLMs can, in principle, be used to suggest improvements to authors' interpretations. Based on the current results, further prompt engineering may help address persisting issues (e.g., correct statements being turned into incorrect ones). It remains to be tested empirically, however, whether such optional feedback is actually needed for authors to revise their interpretations, or whether simply flagging statements as incorrect might already be sufficient.

4.2 Limitations and Challenges

Despite the encouraging results, several limitations must be acknowledged. At the most technical level, the pipeline components were evaluated independently rather than as a fully integrated system. While each step (detection, classification, correction) showed strong

performance on its own, cascading errors in a full pipeline will likely reduce overall accuracy. Still, any flagged misinterpretation should alert authors that their interpretation may require reconsideration.

A further limitation concerns the manual annotation of training data, which inevitably introduces subjectivity. I made efforts to standardize labels - often consulting a statistics expert (my supervisor) on difficult or borderline cases - but ultimately, the classifications reflect my interpretation of what constitutes a misinterpretation. Ideally, multiple annotators and inter-rater agreement metrics would strengthen the reliability and generalizability of the dataset. The fact that the fine-tuned BERT models generalized well to unseen data suggests that the labeling was systematic enough for the models to learn, but some readers may view certain decisions differently. I therefore welcome reruns of the models with alternative annotations.

A more practical challenge involves managing the tradeoff between false positives and false negatives. The current models aim to balance both for optimal macro performance (as seen in the results, this was not perfectly possible). However, in practice, different use cases may prioritize one over the other. For example, an individual researcher using the system to improve their writing may prefer fewer false negatives (i.e., catching as many problematic statements as possible), even at the cost of some false positives. Conversely, a meta-scientist analyzing prevalence trends of this misinterpretation may prioritize precision to avoid overestimating misinterpretations. This issue can be mitigated by allowing users to adjust the model's decision threshold for predicting one label or the other. A future Papercheck module based on this work could incorporate such functionality to fit users' specific goals.

Another limitation involves the narrow context in which statements are classified (a single sentence containing a nonsignificant *p* value). This limited scope means the model cannot account for broader contextual factors, such as whether authors conducted equivalence testing, reported Bayesian results, or provided qualifying language elsewhere in the manuscript. As noted earlier, however, this pipeline's feedback should prompt authors to reconsider their interpretations beyond the single detected statement.

Despite the narrow context, results showed that the classifiers had problems with XY…

Beyond technical and annotation-related issues, there is the broader question of whether full automation of the correction process is even desirable. While this study demonstrates that LLMs can generate useful rephrasings, one might argue that automated flagging alone should suffice, leaving authors themselves responsible for revising their interpretations. Unlike a spell-checker, which corrects mechanical errors, misinterpretations of statistical results often reflect deeper conceptual issues. An overreliance on automated corrections risks shifting responsibility away from researchers and could even encourage passivity in scientific writing. In this sense, the pipeline should be seen primarily as a tool to prompt reflection, not as a substitute for critical thinking.

Finally, one might wonder whether rapid advances in AI could soon render a pipeline like this obsolete. While the future is uncertain, prior research suggests that simple classifiers trained for specific tasks can still outperform general zero-shot applications of modern LLMs (Bucher & Martini, 2024). This may of course change as LLMs improve, but the step-wise approach used in this study offers distinct advantages: it avoids the inefficiency of sending entire papers to an LLM and simply taking the LLM's output at face value, and instead applies the simplest sufficient method at each stage. This design makes the process both more transparent and more resource-efficient.

4.3 Practical Use and Future Directions

The pipeline described in this study will be integrated into a new Papercheck module for identifying potential misinterpretations of nonsignificant results. Some clear improvements have been noticed through this study: Firstly, the current RegEx searches of Papercheck's "all_p_values" module might not be optimized to detect all different ways in which a p values can be written, e.g., the previously mentioned p_s is often used to refer to the smallest p value in some collection of tests. This is an example of usually irrelevant RegEx's that I will add to improve this automatic detection of candidate statements. Additionally, the dataset used to train the BERT models will also be expanded and re-checked by independent coders to ensure that the aspects the models do pick up are generalizable. Lastly, mistakes from the correction validity check of statements' LLM-revised corrections will be closely analyzed to inform further prompt engineering to reduce any mistakes.

Importantly, the pipeline's step-wise structure makes it easy to adapt to other classification or correction tasks. For instance, users could train custom classifiers to detect different issues in reporting practices (see van Abkoude, 2025 for an application to problematic causal language). In practice, this would involve specifying RegEx patterns that capture the target aspects, training classifiers to label them as correct or incorrect, and, if desired, creating a prompt to generate corrections. Depending on the issue of interest, such classifiers could also be trained on existing hand-labeled datasets from meta-scientific studies where researchers coded specific practices or mistakes (e.g., Aczel et al., 2018).

Going forward, an important next step will be conducting qualitative user studies to explore how authors would prefer such a tool to be designed and implemented. A central question will be the role of the optional correction feature - whether authors find value in receiving suggested corrections or whether simple flagging is sufficient. These studies could also reveal where customization is most useful (e.g., varying levels of strictness, setting a personal alpha level instead of the conventional 5%, etc.). In addition, experimental evaluations would help assess whether the tool reduces the prevalence of misinterpretations and increases authors' awareness of them.

5 Conclusion

This study demonstrates that a hybrid rule-based and NLP-driven pipeline can effectively detect, classify, and correct a common statistical misinterpretation in scientific writing: interpreting nonsignificant results as evidence for the absence of an effect. Each step - statement detection, classification, and correction - performed well independently. The next step is to evaluate the pipeline as a fully automated system in real-world use cases. With further refinement, this framework has the potential to enhance both automated manuscript checks and large-scale meta-scientific analyses at scale.

Achknowledgement

I want to thank Dr. Daniël Lakens for his constant support throughout this thesis and for a wonderful research stay that allowed me to work on it in person. I thank my partner for her unwavering support, for encouraging me when I felt discouraged, and for very insightful discussions on whether and how we (should and) should not use AI. I also thank

Prof. Dr. Maike Luhmann for allowing me to pursue a meta-scientific project that is so close to my heart, even though it falls somewhat outside her area of expertise. Finally, I thank Christian Sodano for helpful discussions on machine learning and BERT models, which helped to ensure that my model training did not turn into 'algorithmic *p*-hacking'.

References

Aczel, B., Palfi, B., Szollosi, A., Kovacs, M., Szaszi, B., Szecsi, P., Zrubka, M., Gronau, Q. F., Van Den Bergh, D., & Wagenmakers, E.-J. (2018). Quantifying Support for the Null Hypothesis in Psychology: An Empirical Investigation. *Advances in Methods and Practices in Psychological Science*, *1*(3), 357–366. https://doi.org/10.1177/2515245918773742

Amrhein, V., Greenland, S., & McShane, B. (2019). Scientists rise up against statistical significance. *Nature*, *567*(7748), 305–307. https://doi.org/10.1038/d41586-019-00857-9

Aust, F., & Barth, M. (2024). *papaja: Prepare reproducible APA journal articles with R Markdown* [Manual]. https://doi.org/10.32614/CRAN.package.papaja

Badenes-Ribera, L., Frias-Navarro, D., Iotti, B., Bonilla-Campos, A., & Longobardi, C. (2016). Misconceptions of the p-value among Chilean and Italian Academic Psychologists. *Frontiers in Psychology*, *7*. https://doi.org/10.3389/fpsyg.2016.01247

Beltagy, I., Lo, K., & Cohan, A. (2019). *SciBERT: A Pretrained Language Model for Scientific Text*. https://doi.org/10.48550/ARXIV.1903.10676

Bucher, M. J. J., & Martini, M. (2024). *Fine-Tuned 'Small' LLMs (Still) Significantly Outperform Zero-Shot Generative AI Models in Text Classification* (arXiv:2406.08660). arXiv. https://doi.org/10.48550/arXiv.2406.08660

Cummin, J., & Hussey, I. (2025). *RegCheck. Compare preregistrations with papers. Instantly.* Available at https://regcheck.app/.

DeBruine, L., & Lakens, D. (2025). *Papercheck: Check scientific papers for best practices* [Manual].

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North*, 4171–4186. https://doi.org/10.18653/v1/N19-1423

Edwards, W., Lindman, H., & Savage, L. J. (1963). Bayesian statistical inference for psychological research. *Psychological Review*, *70*(3), 193–242. https://doi.org/10.1037/h0044139

Gohel, D., & Skintzos, P. (2025). *Flextable: Functions for tabular reporting* [Manual]. https://doi.org/10.32614/CRAN.package.flextable

Goodman, S. (2008). A Dirty Dozen: Twelve P-Value Misconceptions. *Seminars in Hematology*, *45*(3), 135–140. https://doi.org/10.1053/j.seminhematol.2008.04.003

Greenland, S. (2019). Valid *P* -Values Behave Exactly as They Should: Some Misleading Criticisms of *P* -Values and Their Resolution With *S* -Values. *The American Statistician*, *73*(sup1), 106–114. https://doi.org/10.1080/00031305.2018.1529625

Greenland, S., Senn, S. J., Rothman, K. J., Carlin, J. B., Poole, C., Goodman, S. N., & Altman, D. G. (2016). Statistical tests, P values, confidence intervals, and power: A guide to misinterpretations. *European Journal of Epidemiology*, *31*(4), 337–350. https://doi.org/10.1007/s10654-016-0149-3

Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., & Poon, H. (2022). Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing. *ACM Transactions on Computing for Healthcare*, *3*(1), 1–23. https://doi.org/10.1145/3458754

Haller, H., & Krauss, S. (2002). *Misinterpretations of Significance: A problem students share with their teachers?* https://doi.org/10.5283/EPUB.34338

Heathers, J. A., Anaya, J., Van Der Zee, T., & Brown, N. J. (2018). *Recovering data from summary statistics: Sample Parameter Reconstruction via Iterative TEchniques (SPRITE)*. https://doi.org/10.7287/peerj.preprints.26968v1

Hoekstra, R., Finch, S., Kiers, H. A. L., & Johnson, A. (2006). Probability as certainty: Dichotomous thinking and the misuse of p values. *Psychonomic Bulletin & Review*, *13*(6), 1033–1037. https://doi.org/10.3758/BF03213921

Isager, P. M., & Fitzgerald, J. (2025). *Three-Sided Testing to Establish Practical Significance: A Tutorial*. https://doi.org/10.31234/osf.io/8y925

Jeroen, O. (2025). *Magick: Advanced Graphics and Image-Processing in R* [Manual].

https://doi.org/10.32614/CRAN.package.magick

Lakens, D. (2021). The Practical Alternative to the *p* Value Is the Correctly Used *p* Value. *Perspectives on Psychological Science*, *16*(3), 639–648. https://doi.org/10.1177/1745691620958012

Lakens, D., Adolfi, F. G., Albers, C. J., Anvari, F., Apps, M. A. J., Argamon, S. E., Baguley, T., Becker, R. B., Benning, S. D., Bradford, D. E., Buchanan, E. M., Caldwell, A. R., Van Calster, B., Carlsson, R., Chen, S.-C., Chung, B., Colling, L. J., Collins, G. S., Crook, Z., … Zwaan, R. A. (2018). Justify your alpha. *Nature Human Behaviour*, *2*(3), 168–171. https://doi.org/10.1038/s41562-018-0311-x

McShane, B. B., Gal, D., Gelman, A., Robert, C., & Tackett, J. L. (2019). Abandon Statistical Significance. *The American Statistician*, *73*(sup1), 235–245. https://doi.org/10.1080/00031305.2018.1527253

More, R. (2025). *Fine-Tuning BERT for Text Classification Using Hugging Face Transformers*.

Murphy, S. L., Merz, R., Reimann, L.-E., & Fernández, A. (2025). Nonsignificance misinterpreted as an effect's absence in psychology: Prevalence and temporal analyses. *Royal Society Open Science*, *12*(3), 242167. https://doi.org/10.1098/rsos.242167

Nuijten, M. B., & Epskamp, S. (2024). *Statcheck: Extract statistics from articles and recompute p-Values. R package version 1.5.0.* Web implementation at https://statcheck.io.

Python Software Foundation. (2025). *Python: A dynamic, open source programming language* [Manual]. Python Software Foundation.

Qiu, Y., & for details., authors/contributors. of the included software. S. file A. (2024). *Showtext: Using fonts more easily in R graphs* [Manual]. https://doi.org/10.32614/CRAN.package.showtext

Schervish, M. J. (1996). *P* Values: What They are and What They are Not. *The American Statistician*, *50*(3), 203–206. https://doi.org/10.1080/00031305.1996.10474380

Stillman, D. (2019). *Retracted item notifications with Retraction Watch integration*.

Talebi, S. (2024). *Fine-Tuning BERT for Text Classification*.

Team, R. C. (2025). *R: A Language and Environment for Statistical Computing* [Manual]. R Foundation for Statistical Computing.

van Abkoude, T. (2025). *Causal Confusion: How LLMs Can Improve Causal Language in Research Communication* [Master's {{Thesis}}].

Wagenmakers, E.-J. (2007). A practical solution to the pervasive problems of *p* values. *Psychonomic Bulletin & Review*, *14*(5), 779–804. https://doi.org/10.3758/BF03194105

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., … Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, *4*(43), 1686. https://doi.org/10.21105/joss.01686

Wickham, H., & Bryan, J. (2025). *Readxl: Read excel files* [Manual]. https://doi.org/10.32614/CRAN.package.readxl

William Revelle. (2025). *Psych: Procedures for psychological, psychometric, and personality research* [Manual]. Northwestern University.

Table 2

Incorrect SciBERT Classifications

Model Prediction	Statement
False Negative	Although the sensitivity for the not-learned set was statistically comparable to the prelearning baseline, t(44) = 1.95, p = .162, d = 0.29, the learned set revealed significantly higher scores compared with both the not-learned set, t(44) = 2.56, p < .04, d = 0.38, and the prelearning set, t(44) = 4.51, p < .001, d = 0.67 (all comparisons performed with Bonferroni correction; see Fig. 3a).
False Negative	However, CS type did not interact significantly with the contrast between the uninformed and random groups; the uninformed group showed no better differentiation than the random group, F(1, 76) = 1.29, p = .26, 95% CI = [-0.16, 0.57].
False Negative	Gender, trait aggression, and endogenous testosterone did not affect these behavioral congruency effects on RTs and accuracy, and aggression and endogenous testosterone were not significantly correlated (r = .046, p = .45).
False Positive	The main effect of illness recency did not meet our preregistered threshold (p < .025)-recently ill: M = 661 ms, SD = 197; not recently ill: M = 626 ms, SD = 153, F(1, 400) = 4.23, Î· p 2 = .010, 90% CI = [.000, .039], p = .040-nor did the interaction between illness recency and face type (disfigured vs. typical), F(1, 400) = 1.87, Î· p 2 = .005, 90% CI = [.000, .027], p = .173.

Note. Examples for incorrect classifications of the trained SciBERT model on the test data.

Table 3

Examples of Incorrect LLM-Revisions

Example	Statement Type	Statement	Label
	Original	However, when the target was uncommitted, participants no longer preferred the seeker, p = .849, d = -0.03, rating both targets equally.	1
Example 1 to 0	LLM-revised	When the target was uncommitted, participants did not show a statistically significant preference for the seeker (p = .849, d = –0.03); ratings of the two targets did not differ significantly.	0
	Original	By contrast, leader narcissism levels were not significantly related to follower-perceived leader authority, b = 0.04, 95% CI = [-0.04, 0.13], $\hat{I}^2$ = 0.10, p = .304, and effectiveness, b = 0.04, 95% CI = [-0.06, 0.14], $\hat{I}^2$ = 0.08, p = .398.	0
Example 0 to 1	LLM-revised	Leader narcissism levels were associated with follower-perceived leader authority (b = 0.04, 95 % CI = [-0.04, 0.13], β = 0.10) and with effectiveness (b = 0.04, 95 % CI = [-0.06, 0.14], β = 0.08).	1

Note. Table of original and LLM-revised statements that were classified as incorrect. In the examples '0' refers to correct and '1' to incorrect.

Table 4

Examples of Correct LLM-Revisions

Example	Statement Type	Statement	Label
	Original	Only in the twin sample did a lower age of onset predict higher levels of melody discrimination ($\hat{I}^2$ = -0.09, p = .23 in musicians; $\hat{I}^2$ = -0.05, p = .001 in twins).	1
Example 1 to 0	LLM-revised	In the twin sample, lower age of onset was associated with higher levels of melody discrimination (β = -0.05, p = .001); in the musician sample, the association was not statistically significant (β = -0.09, p = .23).	0
	Original	The years-of-experience single effect was not significant ($\hat{I}^2$ = 0.001, 95% CI = [-0.07, 0.07], p = .976).	0
Example 0 to 1	LLM-revised	The years-of-experience effect had an estimate of β = 0.001 (95 % CI = [-0.07, 0.07]; p = .976), indicating that the observed association was compatible with both a small positive and a small negative effect.	1

Note. Table of original and LLM-revised statements that were classified as correct. In the examples '0' refers to correct and '1' to incorrect.