How Large Language Models are Transforming Machine-Paraphrased Plagiarism

Jan Philip Wahle[♣*], **Terry Ruas**[*], **Frederic Kirstein**[♠*], **Bela Gipp**[*]
[*]Georg-August-Universität Göttingen, Germany
[♠]Mercedes-Benz Group AG, Germany
[♣]wahle@gipplab.org

Abstract

The recent success of large language models for text generation poses a severe threat to academic integrity, as plagiarists can generate realistic paraphrases indistinguishable from original work. However, the role of large autoregressive transformers in generating machine-paraphrased plagiarism and their detection is still developing in the literature. This work explores T5 and GPT-3 for machine-paraphrase generation on scientific articles from arXiv, student theses, and Wikipedia. We evaluate the detection performance of six automated solutions and one commercial plagiarism detection software and perform a human study with 105 participants regarding their detection performance and the quality of generated examples. Our results suggest that large models can rewrite text humans have difficulty identifying as machine-paraphrased (53% mean acc.). Human experts rate the quality of paraphrases generated by GPT-3 as high as original texts (clarity 4.0/5, fluency 4.2/5, coherence 3.8/5). The best-performing detection model (GPT-3) achieves a 66% F1-score in detecting paraphrases. We make our code, data, and findings publicly available for research purposes.[1]

1 Introduction

Paraphrases are texts that convey the same meaning while using different words or sentence structures (Bhagat and Hovy, 2013). Paraphrasing plays an important role in related language understanding problems (e.g., question answering (McCann et al., 2018), summarization (Rush et al., 2015)), but it can also be misused for academic plagiarism. Academic plagiarism is serious misconduct as its perpetrators can unjustly advance their careers, obtain research funding that could be better spent, and make science less reliable if their misbehavior remains undetected (Meuschke, 2021).

Original Text
... On **April 29, 2017**, **Bill Gates** partnered with Swiss tennis legend **Roger Federer** in playing the "Match for **Africa**" 4, a noncompetitive tennis match at a sold-out Key Arena in Seattle. The event was in support of **Roger Federer** Foundation's charity efforts in **Africa**. ...
Paraphrased using GPT-3
... **Bill Gates** teamed up with Swiss tennis player **Roger Federer** to play in the "Match for **Africa** 4" on **April 29, 2017**. The noncompetitive tennis match at a sold-out Key Arena in Seattle was in support of **Roger Federer** Foundation's charity efforts in **Africa**. ...

Table 1: Example excerpt from a Wikipedia article and its paraphrased versions using GPT-3. Important keywords are highlighted in boldfont and color. Autoregressive paraphrasing with GPT-3 keeps the same message while generating text with the original structure. The original example used is `3747-ORIG-44.txt`.

Paraphrasing tools can be used to generate convincing plagiarized texts with minimum effort. Most of these tools (e.g., SpinBot[2], SpinnerChief[3]) use relatively rudimentary heuristics, such as word replacements with synonyms, and they already deceive plagiarism detection software (Wahle et al., 2022a). However, these tools scratch the surface of the possibilities compared to what large neural language models can achieve in producing convincing high-quality paraphrases (Zhou and Bhat, 2021). Notably, large autoregressive language models with billions of parameters, such as GPT-3 (Brown et al., 2020), make paraphrase plagiarism effortless yet exceedingly difficult to spot.

So far, large language models have found little ap-

[1] https://github.com/jpwahle/emnlp22-transforming

[2] https://spinbot.com/

[3] https://spinnerchief.com/

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 952–963
December 7-11, 2022

plication in plagiarism detection. As language models are already easily accessible for applications such as software development[4] or accounting[5], using language models for machine-paraphrasing will become as easy as a click of a button soon. Therefore, the number of machine-plagiarized texts will increase dramatically in the upcoming years. To counteract this problem, we need robust solutions before models are widely misused.

In this study, we generate machine-paraphrased text with GPT-3 and T5 (Raffel et al., 2020) to compose a dataset for testing against automatically generated paraphrasing. We test different configurations of model size, training schemes, and selection criteria for generating paraphrases. To understand how humans perceive machine-paraphrased text, we also performed an extensive study with 105 participants assessing their detection performance and quality-of-text assessments against existing automated detection methods. We show that while humans can spot paraphrasing of online tools and smaller autoencoding models, large autoregressive models prove to be a more complex challenge as they can generate human-like text containing the same key ideas and messages from their original counterparts (see Table 1 for an example). Popular paid plagiarism detection software (e.g. PlagScan[6], Turnitin[7]) is already deceived by rudimentary paraphrasing methods and large language models make this task even more challenging. We also test the models used for the generation, which show the highest performance in detecting machine-paraphrased plagiarism.

To summarize our contributions:

- We present a dataset with machine-paraphrased text from T5 and GPT-3 based on original work from Wikipedia, arXiv, and student theses to train and evaluate machine-paraphrased plagiarism.
- We explore the human ability to detect paraphrase through three experiments, focusing on (1) the detection difficulty of paraphrasing methods, (2) the quality of examples, and (3) the accuracy of humans in distinguishing between paraphrased and original texts.
- We empirically test plagiarism detection software (i.e., PlagScan) against machine learning methods and neural language models (autoencoding and autoregressive) in detecting machine-paraphrased plagiarism.
- We show that paraphrases from GPT-3 provide the most realistic plagiarism cases that both humans and automated detection solutions fail to spot, while the model itself is the best-tested candidate for detecting paraphrases.

2 Related Work

Plagiarism Detection: Plagiarism describes the use of ideas, concepts, words, or structures without proper source acknowledgment (Meuschke, 2021). Plagiarism datasets are limited to the number of real plagiarism cases known. With the recent success of artificial intelligence in natural language processing (NLP) applications, paraphrase generation and plagiarism detection methods increasingly rely on dense text representations and machine learning classifiers (Foltýnek et al., 2019). Machine learning methods often fail to detect substantial paraphrasing from neural language models (Wahle et al., 2021). In particular, large autoregressive language models (e.g., GPT-3) can generate paraphrased content almost indistinguishable from original work (Witteveen and Andrews, 2019). However, these models are still insufficiently explored in the domain of plagiarism detection, even though their impact on the field is already being discussed (Dehouche, 2021).

Machine-Paraphrase Detection: Machine-paraphrasing can be described as the automatic generation of text that is semantically close to its source and written in other words (Bhagat and Hovy, 2013). Machine-paraphrasing experiences a growing research interest from NLP for learning semantic representations and related applications (Rush et al., 2015; McCann et al., 2018). However, paraphrasing can be used in plagiarism detection to deceive humans and thus needs detection solutions to prevent it (Foltýnek et al., 2019).

Lexical substitution is a common paraphrase mechanism used by plagiarists (Barrón-Cedeño et al., 2013). Many online paraphrasing tools also use synonym replacements and other lexical perturbations to paraphrase text automatically (Foltýnek

[4] https://copilot.github.com/
[5] https://openai.com/blog/openai-api/
[6] https://www.plagscan.com/en/
[7] https://turnitin.com

et al., 2020a). (Foltýnek et al., 2020b) showed that machine-learning classifiers (e.g., Support Vector Machine) could easily detect paraphrasing from popular online paraphrasing tools such as SpinBot. (Wahle et al., 2021) proposed a benchmark with paraphrased examples from autoencoding models (e.g., BERT(Devlin et al., 2019), RoBERTa(Liu et al., 2019)), showing that neural language models can generate more challenging paraphrasing than traditional online tools (e.g., SpinnerChief, SpinBot). In a follow-up study, (Wahle et al., 2022a) evaluate neural language models (e.g., BERT) on paraphrased texts from SpinnerChief, another independent paid online paraphrasing tool. Their main finding was that neural language models outperform machine learning techniques and can obtain super-human performance in all test cases. (Foltýnek et al., 2020b; Wahle et al., 2021, 2022a) results show that synonym replacements are simple to detect with state-of-the-art neural language models. However, none of these studies explore using large autoregressive models in their experiments.

So far, only a few studies have analyzed the impacts of plagiarism using autoregressive models. Seq2Seq models were first used by (Prakash et al., 2016) with stacked residual LSTM networks to generate paraphrases. (Witteveen and Andrews, 2019) train GPT-2 to generate paraphrased versions of a source text and select paraphrased candidates with the highest similarity according to universal sentence encoder(Cer et al., 2018) embeddings and low word overlap when compared to their original counterparts. (Biderman and Raff, 2022) show that GPT-J (Wang and Komatsuzaki, 2021), a smaller version of GPT-3 with six billion parameters, can plagiarize student programming assignments that are not detected by MOSS[8], a popular plagiarism detection tool. The scaling of models allows for the generation of text indistinguishable from human writing (Brown et al., 2020). In addition, the models' increase in size and consequentially their performance (Kaplan et al., 2020) have the potential to make the paraphrase detection task even more difficult.

3 Methodology

This study focuses on understanding how humans and machines perceive large autoregressive machine-generated paraphrase examples. Therefore, we first generate machine-paraphrased text with different model sizes of GPT-3 and T5. We then generate a dataset composed of 200,000 examples from arXiv (20,966), Wikipedia (39,241), and student graduation theses (5,226) using the best configuration of both models.

We investigate how humans and existing detection solutions perceive this newly automated form of plagiarism. In our human experiments, we compare paraphrased texts generated in this study to existing data that use paid online paraphrasing tools and autoencoding language models to paraphrase their texts. Finally, we evaluate commercial plagiarism detection software, machine-learning classifiers, and neural language model-based approaches to the machine-paraphrase detection task.

3.1 Paraphrase Generation

Method: We generate candidate versions of paragraphs using prompts and human paraphrases as examples in a few-shot style prediction (Table 2). We provide the model with the maximum number of human paraphrased examples that fit its context window with a maximum of 2048 tokens total. For both models, we use their default configuration.

Paraphrasing models' goal is to mimic human paraphrases. Instead of manually engineering suitable prompts for the task, we use AutoPrompt (Shin et al., 2020) to determine task instructions based on the model's gradients. As suggested by the authors, we place the predict-token at the end of our prompt. One example of a generated prompt was "Rephrase the following sentence." As humans tend to shorten text when paraphrasing, we limit the maximum number of generated tokens concerning the original version to 90%, which is the approximate ratio of human plagiarism fragments in (Barrón-Cedeño et al., 2013). Table 2 provides an example of the model's input/output when generating paraphrases.

Candidate Selection: Paraphrases that are similar to their source are of limited value as they have repetitive patterns, while those with high linguistic diversity often make models more robust (Qian et al., 2019). The quality of paraphrases is typically evaluated using three dimensions of quality (i.e., clarity, coherence, and fluency), where high-quality paraphrases are those with high semantic similarity and high lexical and syntactic diversity (McCarthy et al., 2009; Zhou and Bhat, 2021). We aim to choose high-quality examples semantically close to

[8] https://theory.stanford.edu/~aiken/moss/

Paraphrase Generation Example
Rephrase the following paragraph while keeping its meaning:
Original: **My day has been pretty good.** Paraphrased: **Today was a good day.**
...
Original: **This paper analyses two paraphrasing methods.** Paraphrased: **We analyze two methods in this study.**
Original: **This text was written by a machine.** Paraphrased: **This sentence has been generated artificially.**

Table 2: Example of generating paraphrased plagiarism with few-shot learning. As input the model receives a **prompt** and human paraphrase **example pairs**. After inserting the **to-be-paraphrased sentence**, the model then generates a **paraphrased version** as the output.

the original content without reusing the exact words and structures (Witteveen and Andrews, 2019).

In this paper, we choose generated candidates that maximize their semantic similarity against their original counterparts while minimizing their count-based similarity. We select the Pareto-optimal candidate that minimizes ROUGE-L and BLEU (i.e., penalizing the exact usage of words compared to the original version) and maximizes BERTScore (Zhang et al., 2019) and BARTScore[9] (Yuan et al., 2021) (i.e., encouraging a similar meaning compared to the original version). Table 3 provides an example for generated paraphrases and their scores. While examples with high count-based similarity usually convey the same essential message (e.g., **Out 1** and **Out 2**), they also share a similar sentence structure and word usage. Examples with high semantic similarity and lower count-based similarity (e.g., **Out 3**) state the same meaning but rephrase the sentence with novel structure and similar words describing the same idea.

Dataset Creation: To provide data for common sources of academic plagiarism (i.e., scientific articles), we paraphrase the original examples of the machine paraphrase corpus (MPC) (Wahle et al., 2022a) which is mainly composed of publications on arXiv, Wikipedia, and student's graduation theses. As human-authored examples, we sample equally from two of the most popular paraphrase datasets, i.e., P4P and PPDB 2.0 (Zhou and Bhat, 2021). The P4P database (Barrón-Cedeño et al., 2013) is composed of realistic plagiarism cases with the paraphrase phenomena they contain (e.g., morphology-based, syntax-based, lexicon-based), and the PPDB 2.0 database (Pavlick et al., 2015) is a large-scale paraphrase corpus extracted with bilingual pivoting from which we extract the high-quality phrasal and lexical subsets.

[9]We use the large model version for both metrics.

3.2 Human Evaluation

Our human study aims to understand how participants perceive machine-paraphrased plagiarism compared to original work and human-paraphrased text. We used Amazon's Mechanical Turk (AMT) service to obtain human assessments for paraphrased text classification. Additionally, we asked experts that actively published in the plagiarism detection domain over the past five years. To have adequate statistical power in our analyses (Card et al., 2020), we included a total of 105 participants (see Appendix A.1 for details on demographic information about participants).

In the first part of the human study (Q2 in Section 4), 50 participants are provided with a mutually exclusive choice of whether a text was machine-paraphrased or original and a text field to justify their reasoning. In the second part (Q3 in Section 4), 50 participants from AMT and five experts from the research community were provided with a mutually exclusive choice of 5 points on a Likert scale for each of the three parameters of clarity, fluency, and coherence. For the first experiment, each participant evaluated five texts for five models resulting in 1,250 text evaluations. For the second experiment, each participant evaluated ten texts for three parameters, totaling 1,340 text evaluations.

Following common best practices on AMT (Berinsky et al., 2012), evaluators had to have over a 95% acceptance rate, be in the United States, and have completed over 1,000 successful tasks. We excluded evaluators' assessments if their explanations were directly copied text from the task (> 90% text match), did not match their classification, or were short, vague, or otherwise non-interpretable. Across experiments, 138 assessments (≈10%) were rejected and not included in the experiments.

4 Research Questions & Experiments

Q1: How does model size influence the quality of generated paraphrases?

A. We ask this question to underline the problem's urgency as recently released models have a large number of parameters. Figure 1 shows the influ-

		BERTSc.	BARTSc.	Rouge-L	BLEU
In:	Later in his career, Gates has pursued many business and philanthropic endeavors.	-	-	-	-
Out 1:	Later, his time was allocated to business and philanthropic endeavors.	0.79	0.74	0.55	0.63
Out 2:	Later in his career, Gates focused on business and charity.	0.84	0.83	0.64	0.51
Out 3*:	**Gates focused on business and charitable efforts later in his career.**	**0.83**	**0.85**	**0.35**	**0.49**

Table 3: Candidate selection of machine-generated paraphrases with an example from (Witteveen and Andrews, 2019). We choose the Pareto-optimal example that maximizes semantic similarity (BERTScore, BARTScore) and minimizes word overlap (ROUGE-L, BLEU). *Selected example in boldface.

ence of model size on the similarity scores of generated candidates against their original candidates on 500 random examples from the PPDB dataset. We test the 220M, 770M, 3B, and 11B versions of T5 and the 350M, 1.3B, 6.7B, and 175B versions of GPT-3 (also known as Ada, Babbage, Curie, and Davinci in the OpenAI API[10] respectively). With the increasing number of parameters, both models' semantic similarity scores (BERTScore, BARTScore) also rise. T5 shows the highest increase when extending the model from 3 billion parameters to 11 billion. GPT-3 (175B) reaches its overall highest semantic similarity, generating sentences with similar meanings compared to the source. Model's generated candidates also have higher count-based scores on average as they often repeat text from the source. As described before, we try to sample candidates with low word-count scores to avoid repetition of words.

We conclude that scaling models' size positively influences their performance at the task of paraphrasing, which agrees with previous research (Kaplan et al., 2020). While the limits and details of scaling models are still unknown, boosting their computing power will allow for more human-like texts to be produced.

Q2. Can humans identify whether a text is original, or machine-paraphrased?

A. This question is inspired by the Turing (1950) Test to differentiate machines from humans. To answer this question, we asked participants to assess whether texts were machine-generated (see Appendix A.3 for more details). We compared original work to an online paraphrasing tool (SpinnerChief), two auto-encoding models (BERT, RoBERTa), and two large auto-regressive models (T5, GPT-3). As examples, we sampled 30 machine-generated paragraphs for each model and their corresponding 30 original texts with an equal weighting between the three sources (Wikipedia, arXiv, and student theses). We performed a Bonferroni-corrected two-sided T-Test to test for statistical significance compared to a control model. As the control model, we chose SpinnerChief with its default paraphrasing frequency as it was the most difficult-to-detect online paraphrasing tool tested in (Wahle et al., 2022a). Participants received individual text examples with three annotation options: "machine-paraphrased", "original", and "I don't know". Participants were not shown aligned examples (i.e., an original and its paraphrased version) to avoid memorization effects.

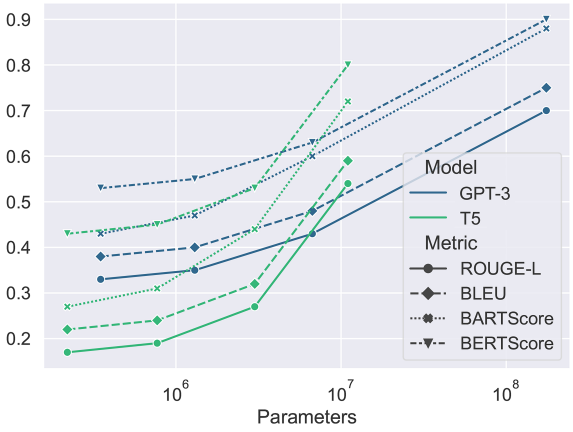

Figure 1: Paraphrasing similarity scores for a sample of the dataset with different model sizes of GPT-3 and T5.

Table 4 shows the mean human accuracy (i.e., the ratio of correct assignments to non-neutral assignments per participant) in detecting machine-paraphrased text. The results show that humans can

[10] https://openai.com/api/

adequately detect the control model with 82% accuracy on average (where 50% is a chance level performance). In contrast, human accuracy at detecting paraphrases produced by autoencoding models was significantly lower, ranging from 61% to 71% over all participants. Plagiarism cases generated by large autoregressive models were usually hardly above chance (53% for GPT-3 and 56% for T5). For more information on the annotator agreement, please see Appendix A.2. Human abilities to detect machine-paraphrased text appear to decrease with increasing model size and are particularly challenging for autoregressive models as they can change sentence structure and word order instead of single word replacements. Our findings on human detection against autoregressive models corroborate with recent results (Clark et al., 2021), challenging the common choice of humans as the gold standard.

Q3. How similar are machine-generated paraphrases to human-paraphrases?

A. We sampled 500 examples pairs (i.e., original, human-paraphrased) from the PPDB corpus and paraphrased half of the original versions with GPT-3 (175B) and the other half with T5 (11B). As a proxy for similarity between originals, human-paraphrased, and machine-paraphrased examples, we calculated their similarity using BERTScore. The average BERTScore between human-paraphrases and originals (76%) is lower than between machine-generated paraphrases and originals (79%). The similarity between human-paraphrases and machine-generated paraphrases is highest (81%). This result suggest that machine-generated paraphrases are typically closer to the human paraphrases than to the original, which we assume is due to the model's objective to mimic human behavior, which are provided as generation examples.

Q4. How do humans assess the quality of machine-paraphrased plagiarism?

A. We asked human annotators to score generated paraphrases according to their clarity, fluency, and coherence (Zhou and Bhat, 2021) (see Appendix A.3 for more details about the questions). As quality assessments are challenging to evaluate, we increased the requirements for participants. We asked the second group of 50 participants that required to have a higher education degree (bachelor's, master's, or Ph.D. degree). We also asked additional five experts that have published at least two peer-reviews papers on plagiarism detection in the last five years. Each participant annotated ten randomly drawn examples on a Likert scale from 1 to 5 regarding clarity, fluency, and coherence (Zhou and Bhat, 2021).

Table 5 shows the average rating for all 55 participants While original contents achieve the highest rating for all three dimensions, the largest version of GPT-3 achieves similar ratings. SpinnerChief's quality of paraphrases is significantly lower. BERT achieves convincing results as well, also because the frequency of word changes (15%) for synonyms is lower than SpinnerChief's (50%), and therefore generates examples closer to the original text.

Fluency was rated highest for all models, while clarity and coherence were the lowest. We assume that as source sentences come from diverse scientific fields, they might already be difficult to understand; thus, paraphrasing can confuse readers when technical terms are used wrong. For more information on annotator agreement and the relation between experts and their educational degree, please see Appendix A.2.

Q5. How do existing detection methods identify paraphrased plagiarism?

A. To test the detection performance of automated plagiarism detection solutions, we evaluate five methods and compare them to random guesses and a human baseline. We presume automated detection solutions can identify paraphrases better than humans as (Ippolito et al., 2020) showed that large language models are optimized to fool humans at the expense of introducing statistical anomalies which automated solutions can spot. As a *de-facto* solution for plagiarism detection, we test PlagScan, one of the best-performing systems, in a comprehensive test conducted by the European Network for Academic Integrity (Foltýnek et al., 2020a). We test a combination of naïve bayes classifier and word2vec (Mikolov et al., 2013), and three autoencoding transformers: BERT (Devlin et al., 2019), RoBERTa(Liu et al., 2019), and Longformer (Beltagy et al., 2020) which are the best performing models in machine-paraphrase detection of (Wahle et al., 2021, 2022a). Additionally, we evaluate the largest versions of T5 and GPT-3 using few-shot prediction.

As paraphrasing models, we choose SpinnerChief; the best performing paid online paraphrasing tool

	Mean accuracy	95% Confidence Interval (low, hi)	t compared to control (p-value)	"I doné know" assignments
SpinnerChief (Control)	82%	76%-89%	-	2.8 %
BERT	67%	63%–71%	14.2 (1e-11)	4.9%
RoBERTa	65%	61%–70%	18.1 (1e-29)	5.5%
T5 11B	56%	51%–59%	16.6 (1e-16)	7.1%
GPT-3 175B	**53%**	**49%–55%**	**19.2** (1e-34)	7.2%

Table 4: Human accuracy in identifying whether parapgraphs of scientific papers from the arXiv subset are machine-paraphrased. Human performance ranges from 82% on the control model to 53% on GPT-3 175B. This table compares mean accuracy of with five paraphrasing models and shows the results of a two-sample T-Test between each model and the SpinnerChief control model according to (Wahle et al., 2022a). Lowest scores are in **boldface**.

	Clarity	Fluency	Coherence
Original	3.98 (±0.78)	4.21 (±0.81)	3.81 (±0.92)
SpinnerChief	2.52 (±1.15)	2.94 (±1.19)	2.83 (±1.23)
BERT	3.45 (±.1.29)	3.34 (±0.90)	**3.73** (±1.22)
GPT-3	**3.92** (±0.97)	**3.60** (±1.02)	3.72 (±1.07)

Table 5: Average scores on a Likert-scale from 1 to 5 of machine-generated plagiarism on the Wikipedia test set. Each example is judged by 50 participants with a bachelor's, master's, or PhD degree and five experts in the plagiarism detection community. Standard deviation is shown in parenthesis. Highest scores are in **boldface**.

tested in (Wahle et al., 2022a). Spinnerchief attempts to change every fourth word with a synonym. We use BERT as an autoencoding baseline and set the masking probability to 15% as in (Wahle et al., 2021). As a large autoregressive model, we use GPT-3 175B, the best model for automated similarity metrics and deceiving humans.

Table 6 shows the average F1-macro except for the human baseline, which shows accuracy. For PlagScan, we assume positive examples when the text-match is greater than 50%. Looking at paraphrased plagiarism of SpinnerChief, humans reach between 79% and 85% accuracy on average. PlagScan achieves results up to 7% over the random baseline for Wikipedia articles but achieves close to random performance for student theses. As in (Wahle et al., 2022a), we assume PlagScan indexes Wikipedia and arXiv but not student theses used in the MPC. Neural approaches based on naïve bayes reach between 58% and 67% F1-macro scores while autoencoding models achieve up to 67% - 78% (Longformer). Large autoregressive models achieve peak scores of 85% (T5 11B) and 87% (GPT-3 175B) on SpinnerChief's paraphrases.

Results of detection models on BERT paraphrasing show similar patterns to SpinnerChief, as autoencoding models also replace masked words with synonyms. While detection results are generally lower for humans and PlagScan, autoencoding models improve by a significant margin. As pointed out in similar studies (Zellers et al., 2019; Wahle et al., 2021), models generating the paraphrased content are typically the best to detect it. The similarity in the architecture of the autoencoding models allows BERT, RoBERTa, and Longformer for the largest performance increase over SpinnerChief. Still, large autoregressive models achieve the best results in detecting machine-paraphrasing of BERT overall, with over 80% F1-score for GPT-3.

When looking at paraphrasing of GPT-3, all models detect paraphrases significantly worse. Humans, plagiarism detection software, and autoencoders can hardly achieve better results than random chance, which underlines how convincing paraphrased texts from large autoregressive models are. T5 and GPT-3 can achieve low, but reasonable results between 60% - 63% (T5) and 64% - 66% (GPT-3) F1-macro.

While detection results on large autoregressive paraphrasing seem low, models were not explicitly trained on the task and are predicted based on previous fine-tuning on other data (upper part) or not fine-tuning (lower part). We assume GPT-3 is the best detection solution because it generated the paraphrased texts. Therefore, we see T5 as a baseline when autoregressive paraphrasing models are unknown.

In general, neural detection models reach their highest performance for Wikipedia articles which we

	SpinnerChief			BERT			GPT-3		
Model	arXiv	Theses	Wiki	arXiv	Theses	Wiki	arXiv	Theses	Wiki
Random	51.72	53.23	49.21	51.90	50.24	48.28	50.61	50.30	49.77
Human Baseline†	83.25	79.32	84.96	68.93	63.41	69.08	55.74	50.60	52.82
PlagScan††	55.07	49.29	57.10	57.73	50.22	59.04	49.28	48.90	50.19
w2v + NB	65.89	58.24	66.83	62.12	59.96	63.38	52.78	51.01	51.15
BERT	64.59	63.59	57.45	80.83	74.74	83.21	52.44	50.89	52.59
RoBERTa	66.00	58.24	58.94	70.41	68.99	72.18	53.14	49.90	53.81
Longformer	78.34	74.82	67.11	65.18	65.72	69.98	54.70	50.84	53.99
T5 11B	**82.92****	**83.45****	**79.92****	**84.66****	**78.09****	**82.37****	**59.80****	**61.42****	**62.72****
GPT-3 175B	**83.20****	**82.11****	**79.68****	**87.21****	**81.02****	**84.48****	**66.52****	**64.38****	**65.79****

Table 6: F1-Macro scores of detection models for text paraphrased by SpinnerChief, BERT, and GPT-3. Numbers in **boldface** are the overall best result. **Results are statistically significant using random and permutation tests (Dror et al., 2018) with $p < 0.05$. †Accuracy calculated as in Table 4. ††F1-score when text-match is greter than 50%.

assume is due to their pre-training data containing Wikipedia examples. Student theses pose the most challenging scenario for both humans and neural approaches, as it contains challenging examples and is written by non-native English as a second language speakers. Across experiments, PlagScan is not able to reliably identify machine-paraphrasing. Large autoregressive models make it challenging for PlagScan to find text matches as phrasal and lexical substitutions can change the words with synonyms and the order of words. The automatic detection results on paraphrasing of GPT-3 are alarming as many of the most used models fail to detect its paraphrases. Even though the absolute results of GPT-3 and T5 are low, they can perform better than humans at the detection task. Therefore, we assume that, similar to (Vahtola et al., 2021), there exist statistical abnormalities and patterns that automated solutions can leverage to increase their detection performance.

5 Epilogue

Conclusion: We generated machine-paraphrased plagiarism using large autoregressive models up to 175 billion parameters convincing paraphrased examples that deceived humans and plagiarism detection solutions. We tested the human ability to detect machine-generated paraphrases of large models and compared their assessments to well-established online tools. We evaluated one plagiarism detection software, one traditional machine-learning model, three autoencoding, and two large autoregressive models detecting machine-paraphrased examples. Despite some limitations, our results suggest that large language models may increase the number of automated plagiarism cases through convincing paraphrasing of original work.

Future Work: This study is an initial step toward understanding how large language models can foster illicit activities in the scientific domain. We plan to further examine the similarities and differences between human- and machine-generated paraphrases to understand whether humans have difficulties in detecting paraphrases in general. When looking at participants' justifications for classifying machine-generated paraphrases, we plan to analyze common terms and highlights to find possible markers for classification decisions. Over the scope of English, our approach could be applied to other languages and even generate paraphrases from one language to another using multilinugal models and data. Finally, as academic plagiarism mainly relies on scientific articles, we want to extend our study to large scientific corpora with high variation across domains and venues (Lo et al., 2020; Wahle et al., 2022b).

Limitations

Although our experiments explore how human and automated solutions struggle to identify machine-paraphrased examples from large language models, we did not detail the similarities and differences between human- and machine-generated paraphrases. Comparing human paraphrases and machine paraphrases - qualitatively and automatically - would allow for a better understanding of what makes paraphrasing so challenging. As the classification from our language models currently does not provide references or sources for their results, these

models can only be used as a support tool to identify sentences and paragraphs for more detailed deliberation. While our study has the above limitations, the focus of this study was to underline the urgency of the problem of machine-generated plagiarism to promote better detection solutions in the future.

Ethics Statement

Plagiarism is illegal, unethical, and morally unacceptable in all countries (Kumar and Tripathi, 2013). While the binary classification of machine-paraphrased examples in this study can indicate how automated detection solutions would point out potential plagiarism cases, a team of experts should make a final decision on such cases. False-positive cases of wrongly accused researchers could ruin their careers forever. Therefore, all cases should be carefully evaluated before any final verdict. As this study and related work show (Clark et al., 2021), humans are unreliable enough for paraphrase detection in the age of large neural language models. The difficulty of machine-paraphrase identification makes legal decisions on plagiarism cases particularly complex. We presume paraphrasing with language models will lead to more plagiarists getting unnoticed when using large models to generate their paraphrases. One exciting approach to gain transparency would rely on reconstructing the model's potential inputs (Tu et al., 2017; Niu et al., 2019) given the paraphrased version and classifying original candidates using a hybrid approach considering text-match and semantic features. We adopted a binary classification in gender for our human evaluation, which we plan to improve in future work so it can be more inclusive. Therefore, gender might not represent the natural diversity included in our dataset.

References

Alberto Barrón-Cedeño, Marta Vila, M. Martí, and Paolo Rosso. 2013. Plagiarism Meets Paraphrasing: Insights for the Next Generation in Automatic Plagiarism Detection. *Computational Linguistics*, 39(4):917–947.

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

Adam J. Berinsky, Gregory A. Huber, and Gabriel S. Lenz. 2012. Evaluating online labor markets for experimental research: Amazon.com's mechanical turk. *Political Analysis*, 20(3):351–368.

Rahul Bhagat and Eduard Hovy. 2013. Squibs: What is a paraphrase? *Computational Linguistics*, 39(3):463–472.

Stella Biderman and Edward Raff. 2022. Neural Language Models are Effective Plagiarists. *ArXiv220107406 Cs*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Dallas Card, Peter Henderson, Urvashi Khandelwal, Robin Jia, Kyle Mahowald, and Dan Jurafsky. 2020. With Little Power Comes Great Responsibility. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9263–9274, Online. Association for Computational Linguistics.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder.

Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A. Smith. 2021. All that's 'human' is not gold: Evaluating human evaluation of generated text. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7282–7296, Online. Association for Computational Linguistics.

N Dehouche. 2021. Plagiarism in the age of massive Generative Pre-trained Transformers (GPT-3). *Ethics. Sci. Environ. Polit.*, 21:17–23.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *ArXiv181004805 Cs*.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1383–1392, Melbourne, Australia. Association for Computational Linguistics.

Joseph L. Fleiss and Jacob Cohen. 1973. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and Psychological Measurement*, 33(3):613–619.

Tomáš Foltýnek, Dita Dlabolová, Alla Anohina-Naumeca, Salim Razı, Július Kravjar, Laima Kamzola, Jean Guerrero-Dib, Özgür Çelik, and Debora Weber-Wulff. 2020a. Testing of Support Tools for Plagiarism Detection. *ArXiv200204279 CsDL*.

Tomáš Foltýnek, Norman Meuschke, and Bela Gipp. 2019. Academic Plagiarism Detection: A Systematic Literature Review. *ACM Computing Surveys*, 52(6):112:1–112:42.

Tomáš Foltýnek, Terry Ruas, Philipp Scharpf, Norman Meuschke, Moritz Schubotz, William Grosky, and Bela Gipp. 2020b. Detecting Machine-Obfuscated Plagiarism. In Anneli Sundqvist, Gerd Berget, Jan Nolin, and Kjell Ivar Skjerdingstad, editors, *Sustainable Digital Communities*, volume 12051 LNCS, pages 816–827. Springer International Publishing, Cham.

Daphne Ippolito, Daniel Duckworth, Chris Callison-Burch, and Douglas Eck. 2020. Automatic Detection of Generated Text is Easiest when Humans are Fooled. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1808–1822, Online. Association for Computational Linguistics.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models.

Ranjeet Kumar and RC Tripathi. 2013. An analysis of automated detection techniques for textual similarity in research documents. *International Journal of Advanced Science and Technology*, 56(9):99–110.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. Cite arxiv:1907.11692.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The semantic scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online. Association for Computational Linguistics.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Philip M McCarthy, Rebekah H Guess, and Danielle S McNamara. 2009. The components of paraphrase evaluations. *Behavior Research Methods*, 41(3):682–690.

Norman Meuschke. 2021. *Analyzing Non-Textual Content Elements to Detect Academic Plagiarism*. Ph.D. thesis, University of Konstanz, Dept. of Computer and Information Science. Doctoral Thesis.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *ArXiv13013781 Cs*.

Xing Niu, Weijia Xu, and Marine Carpuat. 2019. Bidirectional differentiable input reconstruction for low-resource neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 442–448, Minneapolis, Minnesota. Association for Computational Linguistics.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 425–430, Beijing, China. Association for Computational Linguistics.

Aaditya Prakash, Sadid A. Hasan, Kathy Lee, Vivek Datla, Ashequl Qadir, Joey Liu, and Oladimeji Farri. 2016. Neural paraphrase generation with stacked residual lstm networks.

Lihua Qian, Lin Qiu, Weinan Zhang, Xin Jiang, and Yong Yu. 2019. Exploring diverse expressions for paraphrase generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3173–3182, Hong Kong, China. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *ArXiv191010683 Cs Stat*.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Zhaopeng Tu, Yang Liu, Lifeng Shang, Xiaohua Liu, and Hang Li. 2017. Neural machine translation with reconstruction. *AAAI Conference on Artificial Intelligence*.

A. M. Turing. 1950. I.—COMPUTING MACHINERY AND INTELLIGENCE. *Mind*, LIX(236):433–460.

Teemu Vahtola, Mathias Creutz, Eetu Sjöblom, and Sami Itkonen. 2021. Coping with noisy training data labels in paraphrase detection. In *Proceedings of the*

Seventh Workshop on Noisy User-generated Text (W-NUT 2021), pages 291–296, Online. Association for Computational Linguistics.

Jan Philip Wahle, Terry Ruas, Tomáš Foltýnek, Norman Meuschke, and Bela Gipp. 2022a. Identifying Machine-Paraphrased Plagiarism. In Malte Smits, editor, *Information for a Better World: Shaping the Global Future*, volume 13192, pages 393–413. Springer International Publishing, Cham.

Jan Philip Wahle, Terry Ruas, Norman Meuschke, and Bela Gipp. 2021. Are Neural Language Models Good Plagiarists? A Benchmark for Neural Paraphrase Detection. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries (JCDL)*.

Jan Philip Wahle, Terry Ruas, Saif M. Mohammad, and Bela Gipp. 2022b. D3: A massive dataset of scholarly metadata for analyzing the state of computer science research. In *Proceedings of The 13th Language Resources and Evaluation Conference*, Marseille, France. European Language Resources Association.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax.

Sam Witteveen and Martin Andrews. 2019. Paraphrasing with large language models. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 215–220, Hong Kong. Association for Computational Linguistics.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems*, 34:27263–27277.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending Against Neural Fake News. *ArXiv190512616 Cs*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Jianing Zhou and Suma Bhat. 2021. Paraphrase generation: A survey of the state of the art. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5075–5086, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

A Human Study

A.1 Demographic Information of Participants

Participants were given a choice to consent to providing additional anonymous information, including - but not limited to - gender, age, nationality, birth country, current country of residence, first language, and current education level[11]. Out of all 105 participants, 99 provided demographic information. For all participants, we received their total number of completed tasks and the time taken to complete our questions. The average time to rate ten examples was 8.07 ($\pm$ 6.82) minutes. The average number of total successful tasks for participants was 1200 ($\pm$ 590).

The majority of tasks in this study were performed within 3 - 14 minutes (95% of mass in the interval of $[\mu - 2\sigma, \mu + 2\sigma]$). Three participants took significantly longer (23, 27, and 43 minutes), and their ratings were considered outliers on the distribution.

Age & Gender: Participants were 24 years old on average (18 - 41). There was no significant difference in age between men and women with a two-sided T-Test (p=0.87). Figure 2 shows age distribution by gender. The majority of participants were younger than 25 years old.

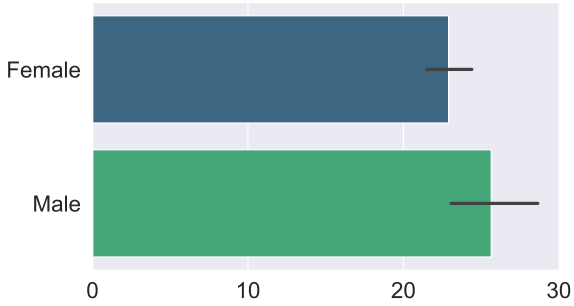

Figure 2: Distribution of age by gender of participants.

Education & First Language: Most participants from *Q4* had a bachelor's degree (68%). The remainder had a master's degree (24%) or Ph.D. degree (8%).

Unsurprisingly, as all participants reside in the US, most of them (78%) had English as their first language. The remainder had Chinese, Spanish, Vietnamese, Russian, or Arabic as their first language.

A.2 Agreement

The inter-annotator agreement according to Fleiss Kappa (Fleiss and Cohen, 1973) of participants for *Q2* was $\kappa = 0.84$.

The inter-annotator agreement of the five experts in *Q4* was $\kappa = 0.66$ and for the remaining 50 participants in *Q4* it was $\kappa = 0.79$.

The agreement between the expert group and the AMT group was $\kappa = 0.41$, showing that experts

[11] The complete list of attributes is available in our dataset.

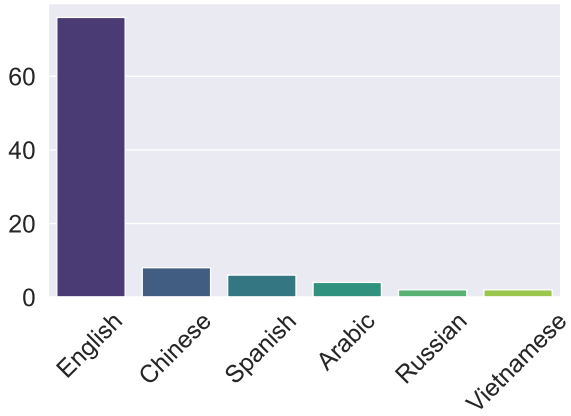

Figure 3: Distribution of first languages of participants.

deviate strongly from average raters with a higher education degree.

When looking at participants with a Ph.D. and a bachelor's degree, assessments of paraphrasing quality deviated more $\kappa = 0.57$ than within the respective groups of participants with a Ph.D. degree $\kappa = 0.79$ and a master's degree $\kappa = 0.77$.

A.3 Details on Questions

For the experiments in *Q2*, participants were asked the following question:

Question: Do you think the above example was machine-paraphrased (which means a machine rewrote some human-authored text) then choose "machine-paraphrased". If you think a human wrote the example, please choose "original". If you cannot assign the example to either category, please choose "I don't know".

For the experiments in *Q4*, participants were given the following three instructions with the option to rate on a scale from one to five.

Instruction 1: The first question is about fluency, which refers to the ability to write grammatically correctly and clearly. Does it sound like a native speaker wrote it (high rating), or does it sound like someone who just learned English (low rating)?

Instruction 2: The second question is about clarity, which refers to the presentation of content and its explanation. Is the content easy to follow (high rating), or is it complicated and hard to understand (low rating)?

Instruction 3: The third question is about coherence, which refers to the consistency of content throughout the paragraph. Is the content following a common central idea (high rating), or is the text jumping from one (random) idea to another (low rating)?