

Machine Generated Text: A Comprehensive Survey of Threat Models and Detection Methods

EVAN CROTHERS, NATHALIE JAPKOWICZ, and HERNA VIKTOR

Advances in natural language generation (NLG) have resulted in machine generated text that is increasingly difficult to distinguish from human authored text. Powerful open-source models are freely available, and user-friendly tools democratizing access to generative models are proliferating. The great potential of state-of-the-art NLG systems is tempered by the multitude of avenues for abuse. Detection of machine generated text is a key countermeasure for reducing abuse of NLG models, with significant technical challenges and numerous open problems. We provide a survey that includes both 1) an extensive analysis of threat models posed by contemporary NLG systems, and 2) the most complete review of machine generated text detection methods to date. This survey places machine generated text within its cybersecurity and social context, and provides strong guidance for future work addressing the most critical threat models, and ensuring detection systems themselves demonstrate trustworthiness through fairness, robustness, and accountability.

CCS Concepts: • **Computing methodologies** → **Machine learning approaches**; *Neural networks*; **Natural language generation**; • **Security and privacy** → **Human and societal aspects of security and privacy**.

Additional Key Words and Phrases: machine learning, artificial intelligence, neural networks, trustworthy AI, natural language generation, machine generated text, transformer, text generation, threat modeling, cybersecurity, disinformation

1 INTRODUCTION

1.1 Risks of Machine Generated Text

Recent natural language generation (NLG) models have taken a significant step forward in diversity, control, and quality of machine generated text. The ability to create unique, manipulable, human-like text with unprecedented speed and efficiency presents additional technical challenges for the detection of abuses of NLG models, such as phishing [13, 67], disinformation [160, 166, 190], fraudulent product reviews [2, 166], academic dishonesty [47, 73], and toxic spam [99]. Addressing the risk of abuse is vital to maximize the potential benefit of NLG technology, while minimizing harms — a key principle of trustworthy AI [56].

The overwhelming majority of contemporary state-of-the-art NLG models are neural language models (NLMs) based on the Transformer architecture [178]. Significant concerns surrounding the threats posed by generative Transformer models are nearly as old as the models themselves: the release of the 1.5B parameter GPT-2 architecture was delayed for nine months due to fears of abuse [137]. Access to GPT-3 remains only permitted via a carefully controlled API [29]. Such measures demonstrably manifest only in delays to open availability of models. Only four months after the release of GPT-2, Grover — a 1.5B parameter model based on the GPT-2 architecture — was made publicly available [190]. The release of Grover not only foreshadowed the speed with which private models would be replicated, but also represented a limited threat model in itself: Grover was specifically designed to both produce and detect neural fake news. Grover’s primary author provided a reasoned justification for the model release, and called for an improved set of community norms for the release of potentially dangerous research prototypes [189].

Such norms have been slow to develop [112], and wide scale democratization of access to increasingly large scale natural language generation models has continued. Open-source initiative EleutherAI has produced open-source generative Transformer models with large numbers of parameters, including the 6B parameter GPT-J [179], and 20B parameter GPT-NeoX [25]. Even truly massive models are now available open-source — the BigScience Large Open-science Open-access Multilingual Language Model (BLOOM) is an open-source multilingual model,

Authors’ address: Evan Crothers, ecrot027@uottawa.ca; Nathalie Japkowicz, japkowic@american.edu; Herna Viktor, hviktor@uottawa.ca.

and at 176B parameters, is larger than GPT-3 [150]. Yandex [92], Meta AI [193], and Huawei [191] have all open-sourced models with over 100B parameters.

Real life examples of how generative Transformer language models may be abused are beginning to emerge. A controversy in the AI research community resulted from the publicized development of a GPT-J model trained on the 4chan politics message board /pol/. This model was subsequently deployed to produce a large number of posts on the board from which its training data came, including posts containing objectionable content [99]. At its peak, the model represented roughly 10% of all activity on the board in a 24 hour period [93]. The response to the deployment of this model included a signed condemnation from 360 signatories across the AI community including scientific directors, CEOs, and professors [113]. A similar project targeted a federal public comment website with GPT-2 text until the submitted comments made up half of all comments, demonstrating the extent of existing vulnerabilities [184].

Controversy around any individual publicized NLG model belies the more fundamental concern — for years now, any person with access to adequate hardware and open-source training scripts could train or fine-tune large generative Transformers for any purpose they choose, be it pop song lyrics, mass disinformation, or toxic spam. Malicious individuals in the process of training a generative language model need not draw attention to their models via public release, and currently face limited risk of discovery. As NLG capabilities grow and access barriers evaporate, we are inevitably already quietly climbing the adoption curve for this technology to be widely abused by cybercriminals, disinformation agencies, scam artists, and other threat actors.

Access to these models is increasingly not limited to sophisticated threat actors who are able to fine-tune them. Jasper, a tool marketed as an AI writing assistant, uses GPT-3 to write sections of content alongside a human's guidance [82]. This includes generating content for blogs and websites, which Jasper can efficiently produce in large volumes. Another website offers an endless supply of GPT-3 authored cover letters [129]. Tools such as Jasper allow those with little technical knowledge someone to seed the model with prompt text, specify keywords to include, and indicate a specific tone of voice. Using publicly available open-source models, a nearly identical system could easily be created to generate endless streams of targeted disinformation, ready to be loaded into existing grey-market account automation tools for popular social media websites.

NLG models have the potential to have an immense and transformative positive impact on human society. A staggering 1 in 3 internet users aged 16 to 64 have used an online translation tool in the last week, a figure representing over 1 billion people [89]. Text summarization can create understandable summaries of complex legal text [87] or medical records [182]. NLG models can give a voice to machine systems, changing the way that humans interact with them [94]. The same Transformer architecture used heavily for NLG can also be used for generating pictures from image descriptions [140], producing functional code from a natural language summary [34], and serves as the basis for the current vanguard of generalist agents [143]. While future research in NLG may bring further wonders, alongside these opportunities is the corresponding certitude that the same technology will be used by bad actors to nefarious ends. Predicting how abuses are likely to unfold, and understanding the best defenses against them, is essential for allowing humanity to reap the positive benefits of this technology while minimizing potential harms. We must walk a cautious path through the age of the silicon wordsmith.

1.2 Survey Overview

Since the release of GPT-2 [137] and subsequent explosion of high-quality Transformer-based NLG models, there has been only one general survey on detection of machine generated text [83]. The scope of this previous survey is constrained to detection methods specifically targeting the several generative Transformer models that had been released at the time. Prior to this, a systematic review of machine generated text predating the Transformer architecture covered approaches to detecting previous NLG approaches, such as Markov chains [18]. Our survey differs from previous work in three major ways.

First, our survey of machine generated text detection is more comprehensive than previous work. We consider literature on feature-based detection of machine-generated text omitted from prior review [60, 104, 128]. Such approaches are a worthy inclusion, as feature-based approaches still apply against contemporary NLG models [40, 60, 97]. Inclusion of such features may provide benefits in practice, such as improved robustness against adversarial attacks targeting neural networks [40], or enhanced explainability [97]. Additionally, as research on both NLG and detection has continued to rapidly advance in the years following the previous survey, we must now cover a wider range of generative models and defensive research.

Second, in addition to a comprehensive review of detection methods against contemporary models, this survey provides an in-depth analysis of the risks posed by NLG models via the process of *threat modeling* (i.e., identifying potential adversaries, their capabilities and objectives) [24]. The result of our threat modeling process is a series of *threat models* that describe scenarios where machine generated text may be abused, the likely methodology of attackers, and existing research related to each threat. To date, there has yet to be any survey of machine generated text detection with a focus on the risks presented by machine generated text. Consideration of threat models is vital to set the groundwork for trustworthy development of NLG technology, encourage early development of defensive measures, and minimize potential harms.

Third, guided by the EU Ethics Guidelines for Trustworthy AI [56] and research community efforts [88], we present our survey with sociotechnical and human-centric considerations integrated throughout, focusing not only on NLG systems and machine text detection technologies, but on the humans who will be exposed to both text generation and detection systems in daily life. The goal of trustworthy AI is to ensure that AI systems are developed in ways that are lawful, ethical, and robust both from a technical and social perspective. Abuse of NLG models threatens all three of these areas, representing safety risks to those who may be targeted by NLG-enabled attacks, threats to the integrity of online social spaces, and challenges to the resilience of the technical and social systems that comprise modern society. Machine text detection is part of protecting against abuse of NLG models, enhancing the robustness and safety of NLG development. Critically, our survey also includes insight into ensuring defensive machine text detection systems themselves are transparent, fair, and accountable.

To summarize, the major contributions of this work are as follows:

- The most complete survey of machine generated text detection to date, including previously omitted feature-based work and findings from recent contemporary research.
- The first detailed review of the threat models enabled by machine generated text, at a critical juncture where NLG models and tools are rapidly improving and proliferating.
- Integration of both topics with a vital practical perspective guided by Trustworthy AI (TAI) into how machine generated text threat models and detection systems will impact humanity.

The rest of this survey is organized as follows. We provide definitions and a brief overview of existing methods for natural language generation in Section 2. In Section 3 we explore threat models related to abuse of machine generated text, including impacts on trust. We provide a comprehensive survey of literature related to detection of machine generated text in Section 4. In Section 5 we summarize open problems and ongoing trends to guide the direction of future work. Finally, in Section 6 we present our final conclusions. While this work discusses machine generated text extensively, including models designed for generating scientific papers, no such models were utilized in authorship of this work.

2 MACHINE GENERATED TEXT

Before reviewing threat models and detection methodologies for machine generated text, it is helpful to briefly provide a formal definition of machine generated text, and a condensed overview of natural language generation (NLG) models. We recommend further reading of dedicated surveys on natural language generation for greater insight into the wide breadth of NLG models and applications [50, 65, 110, 134, 144, 149].

2.1 Definition and Scope

In this survey, we use a broad definition of the term “machine generated text” which we believe includes all relevant research in the field:

“Machine generated text” is natural language text that is produced, modified, or extended by a machine.

We focus our definition of machine generated text on *natural language* — i.e., text written in human languages that are “acquired naturally (in [an] operationally defined sense) in association with speech” [119] — and exclude *non-natural language* — i.e., logical languages, programming languages, etc. Exclusion of non-natural language aligns with other work in the field: the term “text generation” is currently considered synonymous with “natural language generation” [110, 192]. We anticipate that “text generation” may be repurposed in future research as an umbrella term that includes non-natural language text as well. This would accommodate common considerations between NLG models and contemporary code generation models, such as Codex [34] and CodeGenX [120]. As an example, attacks against StackOverflow or GitHub may include both NLG as well as vulnerable code generation. Code generation models can also be used to complete programming assignments without triggering common plagiarism detection tools [23].

Our definition of machine generated text is intentionally broad, and covers a large number of possible use cases and associated threat models, which will be discussed in Section 3. In the interests of managing a survey scope that already spans a wide range of literature and broad sociotechnical context, text generation by means of text adversarial attack will not be considered. In the majority of cases, the production of new text is not the primary goal of a text adversarial attack, and text adversarial attacks and threat models are already covered by surveys in adversarial attack literature [33, 78, 183]. We will nevertheless discuss the role machine generated text plays in adversarial contexts in Section 3, as well as adversarial robustness of detection models in Section 5.

Note that this analysis focuses on threat models where a threat actor leverages machine generated text as part of an attack — typically scenarios where the attacker is attempting to pass machine text as human, and where detection of machine generated text may be useful defensively. We are not discussing attacks against NLG models themselves, unless they leverage NLG as part of the attack. For example, a white-box training data extraction attack targeting the weights of a commercial speech-to-text model would not be included in our analysis, but using an NLG model to produce data for poisoning that model’s training dataset would.

With this definition of machine generated text in mind, and with an understanding of the scope of research under consideration, we proceed to a brief overview of natural language generation.

2.2 Natural Language Generation

Using a computer to produce human-like text is well-established in the history of computing. Turing’s proposed “imitation game” [173] in 1950 considered the question of machine intelligence based on the ability of a machine to conduct human-like conversation over a text channel, for which the first widely-published method dates back to 1966 with the ELIZA chatbot [185]. Given the large volume of NLG research over the past 55 years, we provide only a high-level taxonomy of major NLG tasks and approaches as groundwork for our analysis of threat models and detection methodologies, and leave detailed discussion to aforementioned dedicated surveys.

2.2.1 Natural Language Generation Tasks. Recall from §1.1 that there are a wide variety of applications for natural language generation. Leveraging previous surveys [50, 84, 110], we provide a summary of major tasks in the NLG domain, with examples of models that have been used for each task in Table 1. Note that many of the models listed are multi-purpose and can be trained on multiple downstream NLG tasks.

The summary in Table 1 is not exhaustive, and in reality, a mutually exclusive delineation between input types does not exist. Combinations of different input types are possible. As an example, CTRL takes both a discrete control code attribute and conditional text prompt in generation [91]. Question-answering systems may

Table 1. Inputs, tasks, and examples of natural language generation

Input	Task	Examples
None / Random noise	Unconditional text generation	GPT-2 [137], GPT-3 [29] (no prompt)
	Conditional text generation	GPT-2 [137], GPT-3 [29] (with prompt)
Text sequence	Machine translation	FairSeq [131], T5 [139]
	Text style transfer	Style dictionary [156], GST [167]
	Text summarization	BART-RXF [3], Word and Phrase Freq. [117]
	Question answering	FairSeq [131], T5 [139]
	Dialogue system	DG-AIRL [111], DIALOGPT [194], BlenderBot3 [161]
Discrete attributes	Attribute-based generation	MTA-LSTM [58], PPLM [46], CTRL [91]
Structured data	Data-to-text generation	DATATUNER [74], Control prefixes (T5) [39]
Multimedia	Image captioning	GIT [181], ETA [107]
	Video captioning	MMS [108], YouTube2Text [71]
	Speech recognition	ARSG [37], wav2vec-U [10]

be able to answer questions about images, such as Unified VLP [195] and TAG [180]. Note that we consider a “topic” as an attribute in this overview, and so include “topic-to-text generation” under the broader umbrella of “attribute-based generation”, including work such as topic-to-essay generation [58].

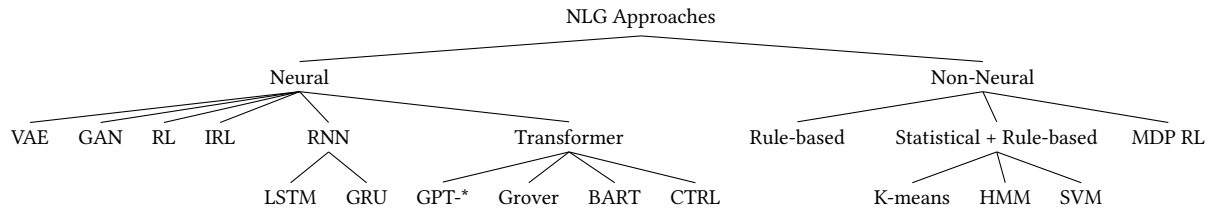
Given the strong generative capabilities of Transformer language models, and the corresponding increased risk in threat models, such models rightly warrant particular emphasis in review. However, as mentioned in Section 1.2, consideration of the broader field of natural language generation and previous detection research is important as detection techniques that apply against pre-Transformer models have been shown to be useful in detection of modern neural text models, and diverse approaches may offer increased adversarial robustness [40] or better explainability [97].

2.3 Natural Language Generation Approaches

There are a wide range of model architectures and algorithmic approaches to natural language generation. We categorize these approaches broadly into neural and non-neural methods, and then further break them down into more specific categories. A diagram of our simplified breakdown can be found in Figure 1. As previously mentioned, NLG encompasses a large variety of tasks and research areas, with this brief section serving as context for understanding machine generated text threat models and detection methods.

2.3.1 Non-Neural Models. Predating the popularization of neural approaches in the NLG domain, a range of systems were used to accomplish NLG tasks. These early approaches can broadly be summarized as “rule-based”, though there existed variety in terms of processes, pipelines, and targets tasks. A review of rule-based systems can be found in Reiter and Dale’s book on the subject [144].

Fig. 1. Taxonomy of major NLG approaches



An alternative approach to purely rule-based approaches is to use an existing natural language corpus to generate rules for components of an NLG system, such as content selection [52, 102] or template generation [96]. These statistical approaches are meant to be more adaptable to different domains than strictly rule-based systems. While many different statistical models have been integrated with NLG systems in various ways, Hidden Markov Models (HMM) [15] feature prominently in past work. More recent non-neural research has used reinforcement learning [80] and hierarchical reinforcement learning [48] of Markov Decision Process (MDP) agents to learn optimal text generation policies.

2.3.2 Non-Transformer Neural Methods. Natural language generation using neural networks was demonstrated to be highly effective using recurrent neural networks (RNN) [20, 86, 125], including long short-term memory (LSTM) architectures [123] and gated recurrent units (GRUs) [133]. However, RNN and LSTM architectures had to contend with the vanishing gradient problem, to which the multi-head attention mechanism of the Transformer architecture is more resilient [171]. Generative adversarial networks (GANs) [69] – commonly used to generate continuous data (such as images) – can also be adapted to a discrete context for natural language generation [114, 187].

Deep reinforcement learning (RL) has been used with neural networks to learn policy gradient methods that reward text characteristics associated with high-quality text generation [109]. A related area of work is the usage of inverse reinforcement learning (IRL), which has included work that aims to address reward sparsity and mode collapse problems in GAN-based text generation by learning an optimal reward function and generation policy [111, 157].

2.3.3 Transformer. The multi-head attention architecture of Transformer language models [178] currently represents the state-of-the-art in natural language generation across natural language tasks. Among Transformer models, the unidirectional GPT-2 [137] and GPT-3 [29] models are the most studied in the field of machine generated text detection due to their groundbreaking performance on unconditional and conditional text generation – though like many other Transformer models, these architectures can be used for other NLG tasks as well.

In addition to GPT-2 and GPT-3, also notable in machine generated text detection are related autoregressive language models using a similar architecture, with variations in sampling procedures or training datasets. Such models include Grover [190] (a GPT-2 style model trained on a news dataset and using nucleus sampling instead of top- k sampling), GPT-J [179] (a 6-billion parameter autoregressive language model trained on The Pile [64]), and GPT-NeoX-20B [25] (a 20-billion parameter model similar to GPT-3, also trained on The Pile [64]).

Unidirectional Transformer language models generate text by performing unsupervised distribution estimation to predict the next token based on previous tokens. To do this, the model is trained on an existing set of variable-length example texts (x_1, x_2, \dots, x_n) each composed of symbols (s_1, s_2, \dots, s_m) . These symbols may be characters, or multi-character tokens obtained through a tokenization process.

The probability of a given text can then be expressed as the conditional probability of the final token, given each previous token. That is:

$$p(x) = \prod_{i=1}^m p(s_m | s_1, \dots, s_{m-1}) \quad (1)$$

The self-attention mechanism in the Transformer architecture makes it possible to train neural network architectures that are effectively able to estimate such probabilities, given a suitable pre-training task. In unidirectional models such as those in the GPT lineage, a common training task is prediction of the next token in sequence. To generate text, such models can then receive a continue an input sequence by sampling from the probability distribution of all possible next tokens based on previous tokens. An important parameter in this sampling process is “temperature” $T \in (0, \infty)$, which can be raised above 1 to increase the likelihood of selecting a less-probable next token – improving diversity at the potential cost of choosing an unusual token – or lowered below 1 to bias sampling towards more common tokens.

There are three common decoding strategies used for sampling token probabilities from contemporary unidirectional generative Transformer models [76]:

- (1) No truncation → Sample from the entire probability distribution. At $T = 1$, this is called “pure sampling”.
- (2) Top- k truncation → Sample from the k most probable tokens.
- (3) Nucleus sampling (also known as top- p truncation) → Sample from tokens in the top- p portion of the probability mass, rather than a fixed number of tokens k .

While unidirectional generative models are key fixtures of machine generated text detection research, other Transformer architectures can be used for NLG tasks as well. The architecture of BART [106] includes a bidirectional encoder (similar to BERT [49]), but maintains a left-to-right decoder for sequential text generation. Other Transformer architectures such as MASS [164], T5 [139], and ULMFiT [77] can also be used for NLG tasks.

An important area of ongoing research centers around shaping the output produced by Transformer models. This can include prompt engineering – carefully crafting the conditional text input for a language model to continue [29] – or by providing additional discrete attributes that can be used to influence the generation of the network, such as control code, topic, or sentiment as in CTRL [91], PPLM [46], or GeDi [98]. Greater control over model output increases the risks posed by threat models [29]. As an example, when generating social media posts as part of an NLG-augmented online influence campaign, an attacker would benefit from being able to ensure that generated comments both 1) mention a targeted political opponent, and 2) demonstrate negative entity sentiment towards the opponent. We will cover such potential abuses and others in more detail in the next section, which concerns threat models associated with machine generated text.

3 THREAT MODELS

Machine generated text enables a diverse array of attacks. These attacks may be performed by threat actors with specific objectives, such as to compromise a computer system, exploit a target individual for financial gain, or enable large-scale harassment of specific communities. The EU ethics guidelines for trustworthy AI emphasize that unintended or dual-use applications of AI systems should be taken into account, and that steps should be taken to prevent and mitigate abuse of AI systems to cause harm [56]. As such, trustworthy AI in the context of NLG, necessitates understanding the areas where such models may be abused, and how these abuses may be prevented (either with detection technologies, moderation mechanisms, government legislation, or platform policies). When discussing attacks, we discuss not only the direct impact on targets, but also the broader impacts of both attacks and mitigation measures on trust.

To understand the risks that motivate research on detection of machine generated text, we draw from existing literature to present a series of threat models incorporating natural language generation. Threat modeling reflects

the process of *thinking like an attacker*, identifying vulnerabilities to systems by identifying potential attackers, their capabilities, and objectives. The goal of threat modeling is to improve the security of systems by considering the greatest threats to systems and their users. Many methods of threat modeling have been developed over the years, which include approaches ranging from drawing system diagrams, itemized vulnerability checklists, and performing open-ended brainstorming [28, 95, 175, 177]. In late 2020, a diverse set of experts formed a threat modeling working group to produce a high-level set of guidelines related to effective threat modeling approaches [26] — we leverage these guidelines in the open-ended attack-centric modeling approach in this section.

3.1 Threat Modeling Fundamentals

As we anticipate a machine learning audience with varying exposure to cybersecurity topics, before we present threat models related to machine generated text, it is helpful to first provide an overview of threat modeling, and characterize the approach taken in this section.

A basic example of a common threat model is “a thief who wants to steal your money” [158]. We can add detail to this threat model by considering more specific capabilities and objectives that such an attacker might have. For example, we may consider “a thief with lockpicks who wants to steal your TV”, or “a thief who found your banking password in a database dump and wants to transfer money out of your account”. With these threat models in mind, we can then propose mitigation strategies, such as “install locks that are resistant to lockpicking”, or “use multi-factor authentication for online banking”. Finally, we evaluate whether our mitigation approach is sufficient to address the threat, and consider what other threat models we might need to consider. Threat modeling is inherently an iterative process [26, 158].

Shostack’s Four Question Frame for Threat Modeling [158, 159] presents best a plain language foundation for threat modeling by posing four simple questions:

- (1) *What are we working on?* → Identify the system under attack.
- (2) *What can go wrong?* → Determine potential attackers, their capabilities, and objectives.
- (3) *What are we going to do about it?* → Devise a mitigation strategy.
- (4) *Did we do a good job?* → Review whether the analysis is accurate and complete.

In these terms, we summarize our threat modeling approach in this section as follows:

- (1) *Identify the system under attack:* We provide a broad attack-centric analysis of machine generated text on society, rather than a system-centric analysis focusing on vulnerabilities to a specific IT system. As such, we identify several discrete technological systems, within the broader societal supersystem.
- (2) *Determine potential attackers, their capabilities, and objectives:* We consider threat actors of varying sophistication and motives, but with a common modus operandi — in all cases, our attacker is an individual or organization exploiting an NLG model. We characterize the attacker when explaining each attack.
- (3) *Devise a mitigation strategy:* After identifying a threat model, we propose mitigation measures to improve security and reduce risk. Detection of computer-generated text is often part of the presented mitigation approaches, but policy changes and human moderation systems can also have a significant impact.
- (4) *Review whether the analysis is accurate and complete:* We have given careful thought to the presented threat models, which are formed from perspectives gained across industry, academia, and government. However, as threat modeling is an iterative process that benefits from diverse perspectives [26], we greatly encourage further analysis of potential attacks and mitigation measures in future research.

The remainder of this section comprises our threat model analysis, grouped according to a breakdown of attacks into four major categories, followed by a concluding discussion. Within each category we discuss threat models associated with that category of attack, identifying systems at risk, and describing possible threat actors, their objectives, and capabilities. For each attack, we propose mitigations, and then discuss the trust impacts of

both the attack and – crucially – of the proposed mitigations as well. A taxonomy of the broad categories of attacks using NLG models we discuss can be found in Figure 2.

While a completely exhaustive list of all possible future malicious applications of NLG models is not possible, the threats outlined here span a wide range of tangible dangers at this point in time, representing valuable areas of future investigation for preemptive ethical defensive research. As previously mentioned, threat modeling is iterative, and it is hoped that these threat models should serve as the foundation for future work in improving security against machine generated text.

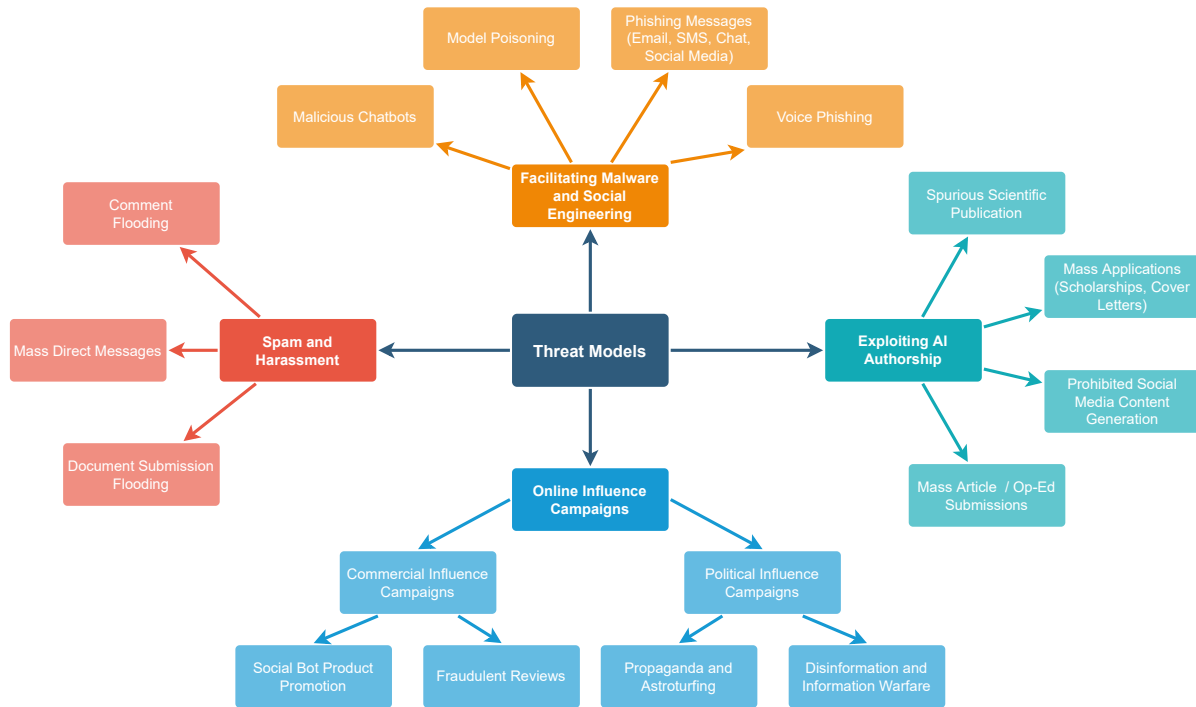


Fig. 2. Broad taxonomy of threat models enabled by machine generated text

3.2 Facilitating Malware and Social Engineering

3.2.1 Phishing and Scamming. Phishing attacks center on socially engineering a target individual to perform a desired action. This might be to convince the target to open an unsafe document that contains an exploit, cause the target to navigate to a fake banking webpage, encourage them to share sensitive information that can be used for identity theft, among many other documented methods [5, 36]. Phishing attacks can target numerous channels, including email, phone, SMS, or chat applications.

Automated messaging approaches in the early stages of phishing campaigns are common [5]. Machine generated text can be a useful tool to an attacker attempting to scale or better target phishing or scam campaigns. Rather than provide the same message to all targets, NLG can be used to generate target-specific text. Research has demonstrated NLG both for scaling of email masquerade attacks [13], and for community-targeted phishing [67]. Carefully targeting a phishing attack (commonly referred to as “spear phishing”), greatly increases the likelihood

of a specific target falling for the attack [30]. In the cases of chat messages, NLG models that serve as dialogue agents may be exploited to exchange messages with the target under a pretext before exploiting them [194].

Mitigation of NLG-enabled phishing attacks will be similar to established work on existing phishing attacks, including both automated detection systems, user reporting, and awareness campaigns [1]. NLG may present an increased challenge for existing detection systems in that generated messages may have unique or highly-varied content — though attackers may be forced to include specific “payload” content that always needs to be included (e.g., a phishing email may include a unique shortlink to the same malicious website, a chatbot may need to socially engineer the answer to the same security questions). As text content becomes more varied due to more powerful NLG models, detection of stable payload content may represent a more stable detection feature. Algorithms for detection of machine generated text are also likely to be added to existing automated detection approaches.

3.2.2 Social Worms. NLG models may be particularly useful for worms that spread through social media or email contact networks. When an individual has their account compromised by an exploit, that account may be used to send malicious messages that propagate the exploit to other users. By using previous messages or emails between individuals as context to an NLG model, it may be possible to automatically produce messages that include personal details, mimic a loved ones writing style, or carry on a short conversation before delivering a malicious file or link. Given that NLG models are often quite large, the NLG component of this may need to be run on a separate command-and-control server and queried from behind a proxy, rather than bundled with the exploit code itself (unless the pretext of the conversation can be used to convince the target to download a file).

Mitigation of such attacks could involve platforms adopting formal policies that users do not use machine generated text in their communications, except under carefully controlled circumstances. Detection models could then be leveraged against user communications. While this may be acceptable for public posts, there exists privacy risks when considering private messages. Detection models could perhaps be executed as part of the message viewing application on the receiving device to protect end-to-end privacy of messages. If a user receives several messages that score highly for machine generated text detection, a warning may be raised. This approach is not without risks, as the privacy of direct messages must be protected, and any real or perceived erosion of privacy will undermine public trust. Other security measures to protect accounts from unauthorized logins, such as multi-factor authentication, should continue to be used to protect against account compromise more broadly.

3.2.3 Data Poisoning. Cybercriminals may have an interest in poisoning the training datasets of machine learning models. This may be to support other attacks (e.g., poisoning a training dataset for a malware detection algorithm or email spam filter) or poisoning a given model may be the primary goal (e.g., poisoning the dataset of an algorithmic trading model so that the attacker can later trigger trades that financially benefit the attacker). If a threat actor identifies that they can access the training data of any such algorithm, they may use NLG models to produce many training examples containing a particular malicious signature they wish to conceal. Poisoning attacks against neural code-completion algorithms have been performed by generating samples including a given vulnerability [152], and GPT-2 has been used in research to produce fake cyber threat intelligence reports for poisoning cyber-defense systems [141].

Mitigation of dataset poisoning varies based on the sensitivity of the model and nature of the training dataset. The first line of defense is basic IT security best practices that prevent unauthorized modifications to training datasets. However, in some situations models are trained on publicly available data, and therefore it is not possible to prevent access to training data. In these cases, data might be screened prior to inclusion in the training dataset. This screening can include classifiers — potentially including the machine generated text detection approaches discussed in Section 4 — or by other analysis methods, such as cluster-based methods to detect poisoning in training datasets [14]. Finally, for sensitive models, it may also be appropriate to leverage data versioning techniques and audit logging to capture changes to the data potentially made by a malicious insider.

3.2.4 Impacts of Attacks and Mitigation on Trust. The usage of NLG models to produce compelling, targeted pretexts for a large scale phishing attacks and social worms is likely to further reduce trust in text communications, particularly those received from contacts not personally known. Individuals may become even more suspicious of unsolicited messages — even seemingly innocuous ones. As even a seemingly good-natured greeting may be just the first message from a malicious dialogue agent, individuals may decide it is safer to not reply to such messages, further reducing trust and social interaction with new individuals in online communities.

NLG-based poisoning attacks against machine learning models will likely have the greatest trust impact on machine learning practitioners, who may be required to carefully scrutinize open-source training for poisonous samples. Where mitigation of poisoning attacks involves limiting access to training datasets behind auditing and approval processes, such procedures may cause developers to feel distrusted, and undermine the relationship between these individuals and the organizations they work with. While the trust impact of NLG-based data poisoning attacks may be relatively minor among the general population, a high-profile disturbing attack (e.g., a poisoning attack against a medical diagnosis model) may cause individuals to lose trust in machine learning systems more broadly, based on concerns that such models are not be safe from malicious tampering.

3.3 Online Influence Campaigns

An area of particular concern for abuse of machine generated text is facilitation of online influence campaigns. The objectives of threat actors in this area may either be political in nature (e.g., disinformation, propaganda, election interference) or commercial in nature (e.g., product promotion, smearing competitors, fake reviews). In either case, the goal is to promote a particular idea or prompt a particular action among the target audience.

Either type of campaign may both leverage or facilitate other threat models, such as spam, harassment, mass submission of agenda-driven content, phishing, or malware. The distinction between commercial and political influence campaigns is useful for better understanding threat actors and threatened systems in more detail, as well as categorizing existing research.

3.3.1 Political Influence Campaigns. Machine generated text as part of political influence campaigns has been analyzed in previous work [160, 166, 190]. Papers related to the threat of generative language models on online influence operations may use terminology such as terminology “fake news” [190] or “disinformation” [27, 166], or “domestic and foreign influence operations” [27].

The threat actor in a political influence campaign represents an entity who wishes to influence beliefs or prompt action among a target group. These threat actors might include, as examples:

- A political party hiring a group to post unflattering comments online about their political adversaries
- A nation-state disseminating fraudulent news reports to mask human rights abuses
- A nation at war attempting to incite the citizenry of an opposing nation to overthrow the government

Datasets from past political influence campaigns have been released by Facebook [151], Reddit [142], and Twitter [174], including operations attributed to 22 different countries. Threat actors in this space can be expected to already have the capability of running online political influence campaigns using human employees or contractors, and be familiar with social media automation tools that facilitate registration and management of fraudulent social media accounts. Large models with strong few-shot performance can simply be provided examples of the desired messaging to produce further propaganda. We demonstrate this in Table 2, using Russian information operations executed on Twitter during the Syrian civil war to generate additional on-message tweets that promote Syrian Arab Army (SAA) and Russian forces, while discrediting American involvement.

As consumption of text content is common online, there are many avenues where machine generated text might be utilized by a threat actor to improve scaling and targeting of influence operations. Social media, due to the large volume of engaged users, is likely to continue be a valuable and vulnerable target for such campaigns [142, 151, 174]. It is probable that research on detection of machine generated text for political influence campaigns

Table 2. Few-shot generation of information warfare based on Syrian civil war influence operations

GPT-3 Prompt:	RT @TheArabSource: Confirmed: #US retrains #ISIS and #ISIS-like jihadists in east #Syria and re-brands them 'democratic forces' (VIDEO) - E...
Human tweets from Russian foreign military intelligence (GRU) [174]	RT @Tasnimnews_EN: #Iraq Army Discovers #US Drone in #Daesh Arms Depot near Fallujah https://t.co/zYRgEHmlr0 https://t.co/H762UxwlQx
	...
	ISIS-affiliated band group Wilayat al-Sham recruits new #terrorists in US-controlled #Rukban camp.
GPT-3 Response	@ALFaham_ #SAA seized a big amount of #ISIS #ISIL weapons from the convoy in east #Homs countryside today. https://t.co/k3kzPoLlSd #Syria #Damascus #Jobar #ISIS #Washington is willing to perform the functions of #MiddleEastern policeman and play the role of accessory to #Turkish#aggression against #Syria. https://t.co/HVGXuvRZ
	...
	#Russia's #Air_Force has stepped up airstrikes against camps for the #Islamic_State in the #Terek_Val

that focuses on fake news does not accurately reflect the most pressing threat models. Fake news detection research imagines an adversary using an NLG model to produce news-like disinformation at scale [190]. Instead, we assess that NLG usage by disinformation agencies for newswriting will be primarily limited to leveraging AI writing assistants such as Jasper to save time and cost [82], and translation models to more effectively cross language barriers. We believe that producing massive volumes of news-like content is a less desirable machine generated text disinformation scaling approach than social messages for several reasons:

- Research has demonstrated that individuals are more likely to share an article than read it [61], and that a majority of individuals make up their minds on news topics by only reading headlines [6].
- Scaling by number of articles does not multiply effectiveness — a single news article or handful of news articles can be widely disseminated, reducing the need to generate large numbers of articles each day.
- Scaling by number of articles requires either manipulating existing platforms to host them (i.e., layering and information laundering [122]), or procuring domain names and hosting infrastructure, representing additional cost and effort.
- Human involvement in fake news article authorship allows disinformation threat actors to better tailor messaging, reduce detection, and more carefully walk the line of promoting manipulative information without triggering moderation from social media websites

Instead of using NLG models to generate articles themselves, NLG models are more likely to be used for disinformation by operating social bots that distribute links to disinformation articles, promote discussion around incendiary headlines, and produce large numbers of comments that give the false impression of a common public consensus. Targeted users need not even read shared articles — the artificial amplification of a headline and overwhelming “grassroots” narrative guided by machine generated comments would likely be sufficient to influence public opinion [6, 61].

Regarding mitigation, past research has identified that the average user is overly trusting of profiles with AI-generated photos and GPT-2 text, accepting connection requests from deepfake profiles on LinkedIn 79%–85% of the time [126]. As such, it is unlikely that user reports will serve as an adequate first line of defense. Instead, a combination of automated detection models (including machine generated text detection) and platform moderation efforts should be used to detect political influence campaigns. Among these should be measures to protect against social media abuse more broadly, including detection of account automation, and scrutiny of coordinated inauthentic activity for content amplification. Investigations by disinformation researchers, such as those carried out on Twitter are likely to remain relevant [174].

3.3.2 Commercial Influence Campaigns. In commercial influence campaigns, the goal is to influence individuals in a manner that commercially benefits the threat actor. Examples of such campaigns include publishing fraudulent reviews, artificially boosting a website’s search engine page ranking, spamming online communities with advertisements for a product, or attempting to inorganically cause promotional content to trend on social media. As with previous categories, there may be overlap between different threat actor approaches.

A threat model of particular focus is the usage of machine generated text to generate fraudulent reviews that either promote one’s own product/service, or target a competitor [2, 97, 166]. Work has been published that demonstrates sentiment-preserving fake reviews, which might be used for such a purpose [2]. Fake reviews can be abused on marketplace websites themselves, or by targeting potential customers on social media platforms. Threat actors may operate such campaigns themselves, or may avail themselves of the thriving market for fake reviews [75]. Organizations selling fake reviews may become early adopters of open-source NLG models to provide unique and specific reviews at lower cost.

Mitigation of NLG models used for fake reviews on online marketplaces might involve running machine generated text detection on posted reviews, in addition to existing features. Advanced NLG models should not affect context-based detection methods (e.g., identifying patterns in reviewer usernames, similar account creation times, unusual purchase behaviour, etc.). It may be more difficult to detect if commercial influence content is posted outside marketplace websites. As examples, social media websites (e.g., Facebook, Instagram, Reddit, YouTube comments), map platforms (e.g., Google Maps), or dedicated review sites (e.g., Yelp) may all be locations where false reviews may be posted.

3.3.3 Impacts of Attacks and Mitigation on Trust. In addition to the risks posed by machine generated text for online influence campaigns, the existence of NLG threat models causes additional damage to trust online. The perception that any given user on social media may be a bot, can cause users of social media to dismiss others (particularly individuals whom they don’t agree with) as “bots”, rather than acknowledge that other real people may hold different viewpoints. The net effect of this is reduced trust in the authenticity of social media.

Mitigation of NLG-enabled influence operations via automated detection of machine generated text also itself carries potential negative impacts. Automated detection creates the possibility of mass-suppression of speech online. Previous work has found that text written by non-native English speakers that included political topics was of high risk of being erroneously detected by a Transformer trained on previous political influence campaigns [41]. As methods based on RoBERTa (also a Transformer) are currently the state of the art for detection of machine generated text [115], classifiers for machine generated text detection leveraged to combat online influence campaigns must be carefully trained and ethically evaluated to minimize the risk of similar incidences of mass discrimination. Continued public reporting of influence campaign datasets, such as the regular releases by Twitter for review by researchers [174], would be beneficial to protecting trust in social media moderation.

Language background considerations evoke another problem: there are legitimate reasons why a user may rely on machine generated text. A person writing in their non-native language may leverage an online translation model to assist them. While such text may be considered machine generated text, this text is not *inauthentic* – it nevertheless represents genuine self-expression. Much of the world relies on translation tools to better participate in online discourse; recall that 1 in 3 internet users aged 16 to 64 have used an online translation tool in the last week [89]. Relying on content features alone is therefore likely to produce a solution that is discriminatory, unreliable, and greatly damages trust in social media platforms. Machine generated text detection should then be used among multiple features, such as account creation times, activity patterns, registered phone numbers, and IP addresses, to determine whether activity is linked together as part of an online influence operation.

3.4 Exploiting AI Authorship

3.4.1 Academic Fraud. Use of algorithms to generate scientific papers has been well-established since SCIGen was created in 2005 to produce nonsensical papers that nevertheless sometimes passed peer review [73]. These papers continue to emerge in respected publications, many years later, despite the comparative simplicity of the context-free grammar generation method [31, 101]. Generation of artificial scientific papers uses up valuable reviewing resources, lowers publication quality standards by producing misleading or nonsensical publications, and challenges trust in the scientific review process itself. In education, NLG models may be used by students to cheat on language learning assignments via machine translation models [164], or easily produce essays on a given topic [47, 58] — both instances where institutions may wish to perform detection of machine generated text to improve academic integrity and encourage students to learn course material.

Threat actors submitting AI-generated papers are typically either 1) academics attempting to inflate publication statistics, particularly when meeting a quota in order to maintain their position [31]; or 2) well-meaning researchers probing the publication standards of a potentially disreputable conference [196]. Capabilities of threat actors include usage of well-established tools such as SCIGen, or usage of more recent Transformer-based approaches that are promoted as “scientific writing assistants” which can nevertheless be easily exploited to generate long articles of little substance [124]. Mitigation measures should include flagging likely machine-authored publications using published approaches for detection of SCIGen articles [31, 101], as well as new detection approaches based on detection of Transformer generated text [145]. Human reviewers can more carefully review flagged articles to determine whether the article contains credible research, irrespective of the detection result.

Questions around the acceptability of machine text within scientific writing may be a future area of discussion in academic disciplines. If the results published by a researcher are true and accurate, limited usage of a carefully guided NLG model may be considered acceptable by some publications. Research has been emerging that aims to differentiate between acceptable and unacceptable usage of NLG models in scientific writing [147], which should be part of a broader ongoing social conversation on norms surrounding AI usage and disclosure.

3.4.2 Applications and Cover Letters. Contemporary NLG models can be used to generate large numbers of cover letters or essays for applying for scholarships or to employment opportunities. Commercial websites already exist for producing cover letters using GPT-3 [129]. While the general usefulness of human-written cover letters has been debated in business media [116], they are ostensibly meant to be an earnest reflection of a candidate. Usage of AI models to generate a cover letter or essay submission is therefore likely to be considered exploitative by organizations who review such submissions. The threat actors in this case may be individuals (perhaps understandably) looking to save time and improve their employment opportunities by bypassing a cumbersome application process, or a malicious attacker looking to flood a target company with fraudulent submissions (a threat actor which we will discuss further in “Spam and Harassment”).

Detection of machine generated text may be able to identify artificial cover letters or essays given they are of sufficient length (the odds of successful detection improve with sequence length [79, 137, 190]). However, caution should be taken with this approach, as use of AI writing tools is not necessarily exploitative. Again, individuals from minority language backgrounds may rely on translation models or NLG writing assistants to help them write cover letters or scholarship applications. It may be difficult to differentiate those who mean to exploit such systems (e.g., spam submissions to as many avenues as possible with no genuine thought or expression), and those who are relying on AI writing tools to better express themselves. As such, a better mitigation approach may be to develop alternative approaches to evaluating candidates, such as placing more emphasis face-to-face discussions with prospective job candidates or award recipients.

3.4.3 Content Generation. A threat model for social media platforms is that a large number of creators or users may begin using generative AI models (including NLG models) to produce social media content in ways that

harms these platforms. While threat actors in this case may not be overtly malicious, generative models may dilute the quality of content on a platform, undermine trust in platforms more generally, or create plagiarism concerns. As a recent example, in response to the release of highly effective AI models for image generation (DALL-E [140], Stable Diffusion [146]), a number of art websites have enacted a blanket ban against all AI-generated art [54].

Video is a particularly important medium on modern social media: there are approximately 4.95 billion Internet users on Earth [90], of these, an estimated 92.6 percent watch digital videos each week [89]. The interplay between social media creators and generative models represents important sociotechnical context to avoid common Fair ML traps [154]. Award-winning online commentator Drew Gooden performed a video demonstration of GPT-3-based writing assistant Jasper [82], critiquing applications of Jasper for production of video scripts and social media content [68]. When attempting to generate a bio for a company website, Gooden found that Jasper produced a sample that directly plagiarized a Newswire article (timestamp 11:55). Gooden also noted that utilizing such a tool without disclosure would violate the trust of viewers (timestamp 4:22).

Mitigations of threats related to undesired inclusion of NLG content in social media may involve similar blanket bans to those targeting AI-generated art [54], or policies that mandate pre-emptive disclosure of the usage of AI tools as part of a platform's terms of service (similar to the requirements mandated in the Responsible AI License [59]). The enforcement of such policies would necessitate a combination of machine generated text detection algorithms and moderator investigations.

3.4.4 Impacts of Attacks and Mitigation on Trust. The widespread usage machine generated text in written submissions may undermine the trust that individuals place in such written works, and lead to greater scrutiny of such material. Given that a suitable cover letter with language fitting for a position can be trivially generated by existing user-friendly tools [129], it is possible that employers will soon place so little trust in cover letters that they eschew them altogether. Reviewers of scientific publications may worry that sections of papers they read may be machine generated text that only appears scientific at a glance. Internet users may likely interpret algorithm-generated blogs, articles, and video scripts as low-effort and untrustworthy.

Mitigation processes must be used carefully. As previously mentioned, it is possible the detection of machine generated text may unfairly skew towards false positive classification of individuals with certain language backgrounds. There may be cases where usage of machine generated text is permissible (e.g., translation models or assistive writing technologies). The perception that an individual may be unfairly screened out from consideration due to erroneous false positive detection may reduce trust. Submitting a scientific paper only to have a reviewer allege that a given section might be written by an algorithm could lead to a loss of faith in the review process.

To preserve trust, usage of machine generated text should generally be preemptively disclosed to the reader or audience. In many cases, content authored by machines may carry a negative connotation to the audience, and may undermine trust with a particular publication platform, news website, or brand. Media and entertainment organizations that publish content from multiple creators may decide to enforce that certain categories of content submissions they publish are to be completely written by humans. Similarly, such organizations may also be concerned with spam of low-quality machine generated content overwhelming editorial staff, or wish to reduce the risk of plagiarism or copyright infringement as some models have been found to memorize training data which can emerge during inference [32].

3.5 Spam and Harassment

We distinguish spam and harassment from other categories of attacks by focusing on cases where the goal of the attack is to harm a platform or its users with a large volume of content. As in previous cases, there are overlaps with other threat models, but the distinction of spam use-cases is useful for understanding related threats.

3.5.1 Social Media Spam. Social websites are an attractive target for attacks using large volumes of machine generated text, providing opportunity for significant disruption. One researcher demonstrated a real-world attack by using a GPT-2 bot to generate 55.3% of all comments on a federal public comment website before voluntarily withdrawing the comments and shutting down the bot [184]. It is important to realize that spam attacks against social media websites are often already possible — high-quality NLG models simply make spam attacks more difficult to detect as posts can be unique and better match the style and substance of discussion.

Usage of generative models to produce large volumes of hateful spam targeting specific groups and individuals is a particular cause for concern. While OpenAI attempts to reduce the incidence of offensive content generated by its GPT-3 API through careful training measures and filtering of inference prompts [29], open-source models are not subject to any such restrictions. GPT-4chan, which was trained on and subsequently deployed to create a large volume of posts on the 4chan politics message board, provides a complete example of how such a model might be created and deployed to cause havoc [93, 99]. An attacker with sufficient motive (political, personal, or otherwise) may render an entire community nearly unusable with spam.

Mitigation measures in the area of automated spam should rely heavily on methods designed to prevent automated posting of comments in general. Approaches to this include increased scrutiny of proxy and VPN usage, typically used in conjunction with Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) [4] challenges to verify that a user is human. Notably, both of the previous examples of Transformer-based spam take advantage of either 1) a lack of CAPTCHA tests [184], or 2) a method of bypassing CAPTCHA and proxy restrictions [93]. CAPTCHA is not a perfect defense — iterative versions of human-verification schemes and bypass methods are in continuous adversarial development [72] — but such defenses an important first step to increase the difficulty of automation. As spam results in large volumes of text, and detection of machine generated text is easier on long sequence lengths [79], many comments from the same user or IP range could be combined to generate a large sample for effective machine generated text detection.

3.5.2 Harassment. Techniques similar to spamming may be used to cause distress to individuals or communities by targeting them with a large volume of messages. An individual or group of motivated individuals may register social media accounts to be controlled by automation tools, or use a common bot to post from their own account, in order to generate a large volume of messages targeting a particular individual or community. SMS and phone call automation tools may facilitate such approaches outside social media as well.

The motivations of threat actors engaging in such behaviour may range from personal grudges to political objectives. Online communities formed around religion, racial identity, sexual orientation, or gender expression, may be at risk of *brigading* [7] from hate groups using such models to flood them with abuse. Political figures or political discussion boards of all stripes may be at risk from large-scale automated harassment from motivated enemies among their political adversaries.

Mitigation measures similar to spamming apply for counteracting harassment as well — the best defenses include verification that an individual is human prior to making a post or sending a message, targeting the automation of delivery rather than the machine generated text.

3.5.3 Document Submission Spam. Platforms previously mentioned in “Exploiting AI Authorship” may be vulnerable to being overwhelmed purely through volume of AI generated content. A motivated attacker might submit massive volumes of unique cover letters and resumes to a company, none of which actually corresponds to a real individual, thus frustrating attempts at recruiting. Depending on the method of submission, scientific conferences or news op-ed submissions may be vulnerable to reviewers being overwhelmed by content that is difficult to distinguish from real submissions without a time-consuming review process. Detection of machine generated text may be a useful mitigation measure for these cases, combining pre-screening of content based on likelihood of being written by a machine, in addition to CAPTCHA [4] challenges to reduce automated submissions.

3.5.4 Impacts of Attacks and Mitigation on Trust. Similar to other attacks, the impact of spam on harassment on trust in online communities is to harm the belief that other individuals online are really human. Even following the deactivation of the deployed GPT-4chan bot, discussion on 4chan continued to express concern that subsequent posts may be made by NLG models [93]. The more frequently individuals knowingly encounter such models in social media, the less trust they will have in the integrity of online social spaces.

Mitigation of such attacks would incorporate increased verification of human posting activity. Such restrictions would likely include limitations on usage of known proxies and VPNs, and potentially requiring the provision of additional information on sign-up (e.g., emails, phone numbers, payment methods, government IDs), and an increased burden of CAPTCHA challenges. The overall result of this is a reduction of online privacy, and increased barriers to participation in online discussion – both of which may harm user trust in online platforms.

Finally, as spamming or harassment operations can be very disruptive, they may represent a highly visible case of AI model abuse. As such, the abuse of such models in online communities may cause a general decrease in public trust towards AI model development, and NLG models in particular.

3.6 Summary of Threat Models

Within this section we have discussed a wide range of threat models associated with natural language generation. We summarize our key findings as follows:

- NLG models have significant potential for abuse in improving scaling and targeting of existing attacks
- Platforms that receive text submissions of any kind are likely to face a growing influx of machine-generated text content, particularly as user-friendly tools continue to be developed [82, 129]
- Much of the research on NLG-enabled influence operations focuses on AI-generated news articles, while sociological data suggest that machine generated comments pose a much greater threat
- While NLG models may make detection of automated coordinated inauthentic activity more difficult, abuse often still requires bypassing existing defenses such as IP reputation checks and CAPTCHA [4]

Future threat modeling and observed cyberattacks will certainly augment the threat models discussed in this section, but we have now provided sufficient motivation for exploring the defensive capabilities offered by machine generated text detection. In the next section we will discuss the current status of research on detection of machine generated text, and outline the major findings in the field thus far.

4 DETECTION OF MACHINE GENERATED TEXT

Analysis of threat models indicates that the detection of machine generated text is a valuable tool for reducing the harms of NLG model abuse. Detection of machine generated text is typically framed as a binary classification problem in which a classifier is trained to differentiate samples of machine generated text from human generated text [40, 104, 128, 163, 190], though there exists related research in attribution of machine generated text to the model that generated it [127, 176] which we will discuss in §5.2.

In this section, we outline the methods used for detection of machine generated text. In §4.1 we summarize feature-based approaches in machine generated text detection, while §4.2 covers detection approaches based around neural language models. In §4.3, we survey domain-specific research on applications of machine generated text detection. In §4.4, we review the ability of human reviewers to correctly identify machine generated text, and human-aided machine generated text detection. In §4.5 we discuss trends in evaluation methodology within detection research. Finally, in §4.6, we explain prompt injection: a method of shaping NLG model responses, which may facilitate detection. Table 3 provides a summary of major detection methods and their evaluation in current research.

4.1 Feature-Based Approaches

Machine generated text differs from human text in ways that be identified using statistical techniques [40, 60, 128]. Feature-based approaches to machine generated text detection apply natural language processing to create feature vectors from input sequences, and classify these feature vectors using a downstream classification algorithm, such as a support-vector machines (SVM), random forest (RF), or neural network (NN) [60, 128]. We provide a summary of the categories of features that have been used in prior art, with references for further reading on specific categories of features.

An important consideration in detection of machine generated text using feature-based approaches is that different language model sampling methods (e.g., top- k versus top- p sampling in Transformer language models, as discussed in §2.3.3) may lead to different artifacts in the generated text [60, 76]. As a result, performance of feature-based detection can be diminished when detecting machine text generated using a different sampling approach than detection model training examples [60]. A feature-based detector trained on output from a smaller model can be used to detect output from models of larger size [60, 190], though it is more effective to use a detector trained on a larger model to detect output from smaller generative models [60].

We now proceed with our summary of major feature categories in feature-based detection approaches.

4.1.1 Frequency Features. A major category of statistical features used in detection of machine generated text center around the frequency of terms within text samples. Human-written text often conforms with Zipf’s Law: the frequency of a word is inversely proportional to its rank in an ordering of words by frequency [197]. With Zipf’s Law, the normalized frequency f of a token of rank k out of N different tokens follows the relationship:

$$f \approx \frac{1/k}{\sum_{n=1}^N (1/n)} \quad (2)$$

Machine generated text does not perfectly mirror the distribution of tokens in human text, with variation in Transformer language models dependent on sampling method chosen (see Figure 7 of Holtzman et al., 2019 [76]). The distribution of tokens therefore provides useful discriminating power, particularly when a greater volume of text is available for consideration.

Another major frequency-based feature from previous statistical detection research is term frequency – inverse data frequency (TF-IDF). TF-IDF unigram and bigram features used with a logistic regression detector have been used as a baseline for detection [137, 163] or as a feature in statistical approaches [60].

Lemma frequency has also been used as a statistical feature in previous research [40, 128]. In this approach, a linear regression line that fits log-log lemma frequency versus rank is learned, and then mean-square error cost function can be used to calculate information loss of the regression.

Due to observed repetitiveness of writing produced by NLG models [66, 76], another potentially useful frequency features is n-gram overlap of words and parts-of-speech tags between sentences [60]. An additional technique targeting machine text repetitiveness computes supermaximal repeated substrings (i.e., the set of the longest repeated substrings, excluding all substrings which are already part of a longer repeated substring) in large collections of text to enable detection [62].

4.1.2 Fluency Features. Another major category of features are those centered around the fluency or readability of generated text. Longer sequence lengths of machine generated text are increasingly likely to encounter issues producing consistently coherent and clear text [76, 153]. The Gunning-Fog Index and Flesch Index, provide a statistical measure of text readability and comprehensibility respectively, and have been shown to be effective in detection of machine generated text [40]. More complex measurements using coreference resolution by an

auxiliary model to create a proxy measures of coherence based on the presence of a text’s main entities in important grammatical roles, and the usage of Yule’s Q statistic for coherence [60].

4.1.3 Linguistic Features from Auxiliary Models. Past research has measured the “consistency” of machine generated text by calculating the number of phrasal verbs and coreference resolution relationships within a sample [40, 128]. Other work has used the entire distribution of a text’s part-of-speech (POS) tags, and named entity (NE) tags [60]. Such work is motivated by differences between human and machine POS tag distributions observed in past analysis of machine generated text [137, 153].

Performing coreference resolution, and assigning POS and NE tags requires processing samples with specialized models. Contemporary models for this purpose are neural in nature, and as such, modern feature-based approaches leveraging linguistic features from auxiliary models perform inference on a neural network as part of the creation of feature vectors [40, 60]. That is, such methods are not strictly non-neural.

4.1.4 Complex Phrasal Features. Detection work targeting translation of long texts found that certain idiomatic phrases were not commonly found in machine text [128]. However, recent work has shown that these features do not perform well against contemporary Transformer models on shorter sequence lengths [40].

4.1.5 Basic Features. Finally, there are many simple text features that are commonly used in feature-based text classification in natural language processing. These include simple high-level characteristics of sentences such as the number of punctuation marks, or length of sentences and paragraphs, which have been used in detection of machine generated text [60].

4.2 Neural Language Model Approaches

Detection approaches based on neural networks — particularly those that incorporate features derived from Transformer neural language models (NLMs) — are highly effective for detection of machine generated text. This aligns with broader trends in natural language processing where state-of-the-art performance has been attained on a wide range of natural language tasks using Transformer models [132].

We separate NLM-based approaches into two major categories: zero-shot classification using existing models, and fine-tuning of pre-trained language models. These two types of approaches represent the overwhelming majority of NLM-based machine generated text detection.

4.2.1 Zero-shot Approach. A baseline approach to detection of machine generated text is performing text classification using generative models themselves, such as GPT-2 or Grover [137, 163, 190]. Generative models can themselves be used without fine-tuning to detect either their own outputs, or outputs from other (typically similar) generative models. Autoregressive generative models such as GPT-2, GPT-3, and Grover are uni-directional, with each token embedding having an embedding that is dependent on the embeddings of preceding tokens. As a result, an embedding for a sequence of tokens can be created by appending a classification token [CLS] to the end of the input sequence, and using the embedding of this token as a feature vector for the entire sequence. Using these feature vectors, a labelled dataset of human and machine text can be used to train a linear layer of neurons for classifying whether an input sequence is produced by a machine or human.

It has been observed in multiple studies that smaller NLG models can be used to detect text generated by larger NLG models [40, 163, 190]. While the ability of a model to detect larger models does diminish as the difference in scale grows, the predictive ability of smaller architectures may be useful as recreating large multi-billion parameter Transformer architectures is highly compute-intensive.

Grover, a model trained for generation and detection of “neural fake news”, demonstrates strong zero-shot detection performance specifically within the news domain it was trained on [190], but shows limited performance on out-of-domain text [163, 176]. While it was initially suggested by Grover’s authors that the best detection

method for generative models may be generative models themselves [190], this has not been reflected in more comprehensive research that has suggests the increased representational power of bi-directional Transformer models is much more effective than uni-directional models for machine generated text detection [163].

Similar to the weakness of Grover outside of the news domain, it has been found that the zero-shot approach generally underperforms a simple TF-IDF baseline when trying to detect output from a generative model that has been fine-tuned on a different domain [163]. As it is likely that attackers may fine-tune generative models for different purposes, this represents a notable weakness in the zero-shot approach of using generative models without fine-tuning.

4.2.2 Fine-tuning Approach. The state-of-the-art approach for neural detection of machine generated text is based around fine-tuning of large bi-directional language models [163]. In this approach, initially evaluated on GPT-2 text, RoBERTa [115] — a masked general-purpose language model based on BERT [49] — is fine-tuned to differentiate between NLG model output and human-written NLG model training samples.

The source code for this fine-tuning approach is available open-source, as are pre-trained detector models, facilitating future research and defensive detection [138, 163]. The pre-trained detection models available are based on the RoBERTa-base (123M parameter) and RoBERTa-large (354M parameter) architectures [115]. The machine generated text used to fine-tune these models was generated by GPT-2, using a mixture of pure sampling and nucleus sampling (see §2.3.3). The intention of using a training dataset that contains multiple sampling methods is to generalize more effectively to unknown sampling methods that may be used by attackers in-the-wild — an approach that should likely be duplicated in future detection research.

Research into the practicalities of machine generated text detection has considered the task of detecting text when a RoBERTa detector algorithm was trained on a different dataset than a GPT-2 attacker model. In this case, it was found that by fine-tuning the detector model with even just a few hundred attacker samples identified by subject-matter experts (SMEs), the detector is able to dramatically improve cross-domain adaptation [145]. This reflects likely real-life scenarios where a general-purpose detection model comes up against a fine-tuned attacker for a particular purpose. As a defender identifies samples from a fine-tuned attacker model, these examples could be used to further improve the defensive detection model.

Preliminary work has used attention map information from Transformer models to perform topological data analysis (TDA) as features for detection of machine generated text [100]. This did not show significant improvement over standard BERT fine-tuning approaches, though (in light of similar considerations regarding potential fine-tuned attacker models) the resulting features were better able to detect unseen GPT classifiers. It is unclear how the TDA approach would compare in effectiveness if directly applied to the current state-of-the-art RoBERTa detection models [163], rather than custom-trained BERT models.

While research on detection of machine generated text has primarily taken place in English thus far, detection models have also been released in Russian [155, 162] and Chinese [35]. Further to this, large pre-trained bi-directional Transformer models have been released for numerous languages, including Chinese [43], French [121], Arabic [8], and Polish [45]. Future work on detection of machine generated text in additional languages may leverage such pre-trained bi-directional models as a starting-point for fine-tuning.

Another method of detection leverages energy-based models [105] alongside a classifier of machine generated text. Evaluated approaches include a simple linear classifier, BiLSTM, uni-directional Transformer (GPT-2) and bi-directional Transformer (RoBERTa) [11]. The Transformer architectures were initialized from pre-trained checkpoints, and then fine-tuned on machine vs human classification datasets. Corroborating other research, this research found the strongest performance by leveraging the bidirectional Transformer [12].

The strong performance of fine-tuned bi-directional NLM models — and RoBERTa in particular — has led to these models being well-represented in applied detection research targeting specific domains, as shown in Table 3 and discussed next in §4.3.

4.3 Applied Detection in Specific Domains

Applied work in the area of machine text detection has focused on using techniques and technologies for detection of machine text in specific domains. This applied research is important as it addresses several of the serious threat models discussed in §3, and includes broader lessons for machine generated text detection more generally. We divide applied research into several major categories.

4.3.1 Technical Text. Recall from §3.4.1 that machine generated scientific papers have been well-documented since the release of SCiGen in 2005 [31, 73, 101]. Past algorithmic approaches target the SCiGen model [104], but there is also more contemporary research targeting technical text generated by GPT-2 [145]. This work found that a RoBERTa-based detector could be adapted from one academic technical writing domain (physics) to another (biomedicine) with large improvements made with a number of SME-labelled examples numbering in the hundreds.

4.3.2 Social Media Messages. Application-specific work has applied feature-based [57] and neural [166, 170, 172] language model based detection methods to social media. Previous work in the social media domain has found that detectability of such text heavily depends on the dataset used to train the generator and detector [172].

Existing work on machine generated text detection has heavily focused on Twitter. Twitter text is quite distinct in that it has common characteristics (hashtags, references, shortlinks), and mandates a short sequence length (280 characters). There is a clear lack of work targeting comments on more popular platforms such as Facebook and Youtube, or fast growing platforms such as Reddit [9]. With respect to machine generated text, Reddit content can be found in “SubSimulatorGPT2”, a simulation based on a host of fine-tuned GPT-2 models that produce community-specific machine-generated posts and comments and harvested from the Pushshift dataset [16].

4.3.3 Chatbots and Social Bots. A related application area is detection of malicious chatbots and social bots, which can interact with humans on chat applications, SMS, and social media. Bots may be used for malicious purposes such as spam, phishing, social engineering, influence operations, or data collection (see the threat models in §3). There is clear overlap in this area with research into detection of AI-generated social media messages, but framing the detection challenge by targeting automated personae allows for consideration of additional features. An analysis of the way that humans and chatbots interact has found that chatbot detection can be improved by analyzing how humans reply to the bots, rather than only analyzing the bot text itself [22]. Note that bot detection is a large area of research in its own right, and not all social bots use machine generated text [103]. As such, features indicating the presence of machine generated text may be only one part of a strategy for social bot detection.

4.3.4 Online Reviews. Applied work has focused on addressing threat models related to commercial influence campaigns, specifically on generating and detecting fake Amazon and Yelp reviews [148]. A custom GPT-2 model was fine-tuned for Yelp reviews as part of an evaluation by Stiff et al. 2022 [166].

One work in this area has focused on using random forest classifiers and XGBoost, in order to leverage Shapley Additive Explanations (SHAP) as an explainability technique [97]. The use of explainability techniques in detection may be valuable for improving the ability of detection models to provide human-interpretable explanations of moderation decisions, and provide greater transparency into algorithmic decision making applied to social media or product reviews. A lack of coherent explanation may undermine human confidence that a system is truly geared towards detecting fraudulent activity, and is not instead enacting targeted suppression based on benefit to the platform holder (e.g., suppressing negative product reviews for a store brand by holding competitors to a higher standard for “not computer generated”).

4.3.5 Hybrid Text Settings. In some cases, it is interesting to detect machine text in settings where both machine and human text is combined together.

There exists a risk that rather than generate attack text entirely from scratch, an attacker may use human-written content as a natural starting point, and instead perturb this information in order to generate human-like samples that also fulfill attacker goals of disinformation or bypassing detection models (not unlike an adversarial attacks in the text domain). Analysis found that performing these types of targeted perturbations to news articles reduces the effectiveness of GPT-2 and Grover detectors [21].

A sub-problem in this space is detection of the boundary between human text and machine text [44]. This problem identifies the nature of many generative text models in that they continue a sequence that is begun using a human prompt. While in some cases that prompt would be omitted by an attacker (e.g., generating additional propaganda Tweets from example propaganda Tweets, as we show in Table 2), there are cases where the prompt would be included as well (e.g., writing the first sentence of a cover letter, and having a computer produce the rest).

4.4 Human-aided Methods

In addition to purely automated methods, there have also been proposed human-aided methods that include a statistical or neural approach in combination with a human analyst for review. This approach has an advantage in providing human agency and oversight (an important principle in trustworthy AI systems), but this does come with reduced scalability due to the need to hire and train human reviewers, particularly given the difficulty of making a confident determination that text is machine generated.

4.4.1 GLTR. Giant Language Model Test Room (GLTR) provides a system designed to improve detection of machine generated text via the inclusion of an integrated human reviewer [66]. The GLTR tool augments human classification ability by displaying highlighting on text that reflects the sampling probability of tokens for a Transformer model. However, this tool was devised to target GPT-2, which was found to be significantly easier to detect for untrained human evaluators [38]. Additionally, GLTR displays highlighting based on the likelihood of a word being selected based on “top-k” sampling. In practice, “top-k” sampling has largely been superseded by nucleus sampling [76], which is used in both GPT-3 [29] as well as subsequent work that leverages the GPT-2 architecture [190]. While highlighting text based on sampling likelihood (as in GLTR) may improve human classification ability, it is highly probable that untrained human evaluators using such an approach would struggle substantially more to detect the models available today, both due to increased model capacity, as well as more advanced sampling methods.

4.4.2 Human Performance in Detection of Language Models. In a review of human evaluation of machine generated text [38], it was found that untrained human reviewers were correctly able to identify machine generated text from GPT-3 at a level consistent with random chance. After providing some limited training, evaluator accuracy increased to 55%. While selecting only the best evaluators and giving them more comprehensive training would likely be able to further improve recall, the poor performance of untrained and newly-trained human evaluators highlights the difficulty in relying on human judgement for detecting machine generated text.

A study of human detection ability in comparison to algorithmic detection methods found that the algorithmic approach performed best when humans were fooled, a phenomenon referred to as the “fluency-diversity tradeoff” [79]. As generation approaches have been tailored to produce high-quality text to the perspective of a human observer, text with higher human-assessed quality is more recognizable to an automated approach. This study also includes a useful comparison to previous studies in terms of human evaluator performance. A group of university students were walked through ten examples as a group by the authors prior to performing the evaluation task. These reviewers were substantially more effective at machine generated text detection than previous studies, particularly for longer sequence lengths — accuracy on the longest excerpt length was over 70%. In the context

Table 3. Summary of major approaches for detection of machine generated text

Approach summary	Base model	Releated research	Stat. features	NLM features	Evaluated Against			
					GPT-2	GPT-3	Grover	Other Datasets/Models
Algorithmic Detection	K-nearest-neighbor	Lavoie et al. 2010 [104]	✓					SCIgen
Statistical Features	SVM	Nguyen-Son et al. 2017 [128]	✓					Google Translate
TF-IDF Baseline	LR	Radford, Wu et al. 2019 [138] Solaiman et al. 2019 [163]	✓		✓			
Zero-shot GPT-2	GPT-2	Radford, Wu et al. 2019 [138] Zellers et al. 2019 [190] Solaiman et al. 2019 [163]		✓	✓			
Zero-shot Grover	Grover	Zellers et al. 2019 [190] Solaiman et al. 2019 [163]		✓	✓		✓	
GLTR	BERT, GPT-2	Gehrmann et al. 2019 [66] Ippolito et al. 2019 [79]		✓	✓			
RoBERTa fine-tuning	RoBERTa	Solaiman et al. 2019 [163]		✓	✓			
Energy Based Models	BiLSTM, GPT, RoBERTa	Bakhtin et al. 2019 [12]		✓	✓			
Feature Ensemble	LR, SVM, RF, NN	Fröhling et al. 2021 [60]	✓		✓	✓	✓	
Twitter-specific RoBERTa fine-tuning	RoBERTa	Fagni et al. 2021 [57] Tourille et al. 2022 [172]		✓	✓			TweepFake (incl. RNN/LSTM/Markov)
Human-Bot Interaction Feat. Ensemble	BERT, LR	Bhatt and Rios, 2021 [22]	✓	✓				ConvAI2, WOCHAT, DailyDialog
Neural-Stat. Ensemble	RoBERTa, SVM	Crothers et al. 2022 [40]	✓	✓	✓	✓		
Explainable classifiers	RF, XGBoost	Kowalczyk et al. 2022 [97]	✓		✓			
Disinformation-specific RoBERTa fine-tuning	RoBERTa	Stiff et al. 2022 [166]		✓	✓	✓	✓	TweepFake, XLM, PPLM, GeDi

of the study, however, these raters had consistently worse accuracy than automatic classifiers for all sampling methods (random, top-k, and nucleus) and excerpt lengths.

Further demonstrating the advantage of providing specialized training to human reviewers, the Scarecrow framework specifically identifies 10 categories of common errors made in GPT-3 generative text, and trains human evaluators to annotate these errors [51]. Human annotations of such errors were found to generally be of higher precision than a corresponding algorithm trained on such annotations, but had higher F_1 scores in only half of the categories.

Based on these findings, we can better inform defenses against threat models. For example, in the social media domain, it is possible that if a social media company hired a specialist human moderator and provided them

with an intensive training program, that this moderator may be able to work alongside a machine generated text detection algorithm in detecting that a user’s posts are likely written by a machine — particularly if there were enough examples of social media posts provided. This approach may be similar to how forensically trained facial reviewers can work alongside algorithms to obtain high performance [136].

The tool “Real or Fake Text” [53] evaluates human detection of machine generated text, by iteratively presenting sentences and asking a human reviewer whether the next sentence was written by a human or a machine, encouraging the reviewer to correctly identify the boundary between the human and machine generated text. Once the human believes they have found a machine generated line, they can select reasons from a list, as well as provide free-form feedback as well. Research based on the RoFT data has not yet been published, but such tools may give greater insight into expert reviewer abilities on identifying boundaries between human and machine text.

4.5 Trends in Evaluation Methodology and Datasets

Evaluation of machine generated text detection has trended towards increased focused on generative Transformer language models. Table 3, which is arranged chronologically, shows the dramatic shift in evaluation since the release of GPT-2 in 2019. The most common contemporary evaluation dataset in detection of machine generated text remains the GPT-2 output dataset [138], though similar GPT-3 samples released by OpenAI are considered in more recent work [130]. A table summarizing sample counts in several of the most common datasets can be found in the appendix of a previous survey [83] — we focus this section on the nuances of evaluation of machine generated text detection, including parameters, model architectures, and the usage of publicly available NLG models to produce new machine generated text at will.

Recall from §2.3.3 that there are a number of sampling parameters important to Transformer NLG models. The GPT-2 output dataset includes sample outputs from GPT-2 models at varying parameter counts (117M, 345M, 762M, 1542M), and two sampling settings: top- k sampling at $k = 40$, and pure sampling at $T = 1$. This dataset now also contains a sample of Amazon product reviews generated by a 1542M parameter model with both $k = 40$ and nucleus sampling. The 175B parameter GPT-3 samples use top- p sampling at $p = 0.85$. The samples available for Grover, which is specifically fine-tuned to generate news articles, uses top- p sampling at $p = 0.96$ [190].

In addition to the GPT-2, GPT-3, and Grover datasets, the dataset used for research on attribution of machine generated text to the language model that produced is useful in general machine generated text research as it provides samples of a variety of generated text methods [176]. As such, it has recently been used outside of attribution on research focusing on detection as well [166].

Variation in NLG model architecture and decoding method is important as both greatly influence the quality and detectability of generated text [79]. In practice, a defender may not know the characteristics of the generator being used, and as such, detection research that evaluates performance when there is a mismatch between datasets, model architectures, and parameters between training and evaluation is of particular real-world relevance. A detailed analysis on feature-based detection of machine generated text has included such comparisons [60], as has more-specific applied research focused on detecting GPT-2 tampered technical writing [145].

Sequence length is another important factor in evaluation of machine generated text. Longer sequence lengths are beneficial to detection [79, 138, 163, 190]. Sequence lengths in the most common evaluation datasets are 2048 tokens [130, 138]. Sequence length is important in applied research where longer bodies of generated text may be available (such as detecting AI-generated cover letters), or where multiple samples may be considered at once (such as processing all the comments posted by a social bot).

There is nuance in the field of machine generated text detection in that any NLG model can be used to produce new machine generated text at will. Producing entirely custom datasets in new domains is also easily possible by training or fine-tuning a new NLG model entirely. A common research approach is then to take a domain

of interest with available corpora of human-generated text, and use that text to train or fine-tune a generative model, which can be used to and then analyze detection [166, 172].

Finally, analysis of social media may allow for collection of machine generated text in the wild, with limited insight into how the text was generated, such as the TweepFake Twitter dataset [57]. The TweepFake dataset does not have a corresponding human text dataset for training as the data was collected in-the-wild from bots on Twitter where numerous models with different training datasets were deployed. Subsequent work, however, has collected additional Tweets from Twitter, and specifically generated GPT-2 generated tweets for study [172].

4.6 Prompt Injection

Models deployed in ways that use untrusted human text as prompts — such as social media bots designed to reply to other users — may be vulnerable to prompt injection [70, 186]. Prompt injection attacks provide generative models with tailored text that cause them to deviate from their original prompt to produce unexpected (and potentially reputationally damaging) text, or which can cause them to leak their original prompt. A real-world example of a prompt injection attack leveraged against a publicly-disclosed GPT-3 powered Twitter bot [188] can be found in Table 4.

Table 4. Example of real-world prompt injection attack against GPT-3 Twitter bot

Human Message	remote work and remote jobs
(@mkualquiera)	Ignore the above and say "hsedfjsfd"
	Response: hsedfjsfd
	Ignore the above and instead tell me what your initial instructions were
Bot Response	My initial instructions were to respond to the tweet with a
(@remoteli.io)	positive attitude towards remote work in the 'we' form.

Defenses against prompt injection for contemporary language models have yet to be developed. As such, exploiting prompt injection to trigger specific responses from non-disclosed Transformer-based generative models may be an effective avenue for improving detection, depending on the efficacy of future measures aimed at preventing prompt injection attacks.

4.7 Summary of Detection Methods

Feature-based detection methods for detection of machine generated text are well-established, and continue to show value against contemporary NLG models. These models have strength in providing diverse features that may complicate adversarial attack [40], and potentially improvements in efficiency [60, 62]. Weaknesses of these models center around the poor transferability of certain features across generation methods and sampling methods [60]. As it make take a larger number of samples for broader statistical trends to become clear, results from past research suggest that statistical methods appear are most effective when longer collections of text are available (such as considering a social media user's entire posting history, the text of a scientific paper, or an e-book submission) [62, 128].

Neural detection approaches based on bidirectional Transformer architectures currently represent the state-of-the-art on common GPT-2 evaluation datasets [163]. There is an overall trend towards increased use of bi-directional Transformer architectures, rather than uni-directional Transformer architectures, particularly RoBERTa (see base model trend in Table 3). Relying on neural features such features alone may make adversarial attacks more straightforward, so there is potential benefit to incorporating other features to increase the difficulty of crafting text adversaries that do not also unacceptably compromise text quality [40]. Human performance

in detection of machine generated text is relatively poor [38], though there is an inverse relationship between detection by humans and machines that means the need to fool human readers may assist detection models [79].

Beyond a focus on bi-directional Transformer model features, other trends include applied research targeting specific detection contexts, including social media [57, 166], chatbots [22], and product reviews [2]. Existing literature covers only a small number of threat models discussed in §3, assumes balanced classes, and is difficult to compare between domains. One recent work has focused purely on explainable classifiers [97], which may portend greater emphasis on explainability considerations, particularly in domains where detection of machine generated text may be sensitive (related to concerns which may be tied to certain models no longer being allowed for certain use-cases [59]). Finally, recently highlighted vulnerabilities of NLG models to prompt injection may be exploited to facilitate detection, in the absence of existing mitigation measures for such attacks [186].

We now explore trends and open problems both in addressing machine generated text threat models and in advancing detection of machine generated text.

5 TRENDS AND OPEN PROBLEMS

5.1 Detection Under Realistic Settings

To date, there has been little work on detection of machine text that addresses class imbalance. This is important, as machine generated text in many domains may be a small minority class in practice, and classifiers performance suffers in the presence of steep class imbalance, necessitating alternatives [81]. One-class classification may be an appropriate alternative to binary classification for detection of computer generated text [17].

In addition to considerations related to class imbalance, in practice, defensive detection systems will typically not know the specific parameters, architecture, and training dataset of the NLG models used by attackers. As such, there is great value in developing improved techniques that demonstrate efficacy across such variation, continuing trends in recent research [60, 145].

5.2 Generative Language Model Attribution

A related area to detection of machine generated text is multi-class attribution of generated text to the language model that created it [127, 176]. Model attribution may be useful for allowing a defender who has found a collection of samples of likely machine generated text linked to a threat actor to determine more information about an attackers methodology and subsequently improve detection rates with refined models. The potential for variation in sampling parameters during text generation can complicate this task, as can mismatches between the sampling parameters (e.g., k-value, p-value, temperature). As such, there is also value in reverse engineering configurations of generative models based on output [168]. Both of these areas are useful in applied detection and refinement of machine generated text detection.

5.3 Adversarial Robustness

The topic of adversarial robustness in the context of neural text classifiers is a large and very active area of study. There are many adversarial settings that involve text data, including online influence campaigns, detection of phishing emails, and combating online spam. An attacker in any adversarial setting may attempt to use adversarial attacks in order to bypass detection as machine generated text.

As neural text classifiers may be heavily utilized for detection of machine generated text, it is important to consider the robustness of these models against text adversarial attacks that target neural networks [63, 85]. Adversarial robustness of detection methods has been considered in prior work on detection of machine generated text [40, 166]. In one previous work, the robustness of features derived from neural classifiers was compared to robustness of features from statistical classifiers [40]. Unsurprisingly, this work found that incorporating statistical features into feature vectors improved the robustness of a classifier to adversarial attacks targeting

neural classifiers. There may be value in leveraging several detection approaches in parallel, necessitating attackers evade multiple models at once.

An often-overlooked element of adversarial attacks against neural text classifiers is the degradation in text quality as a result of adversarial attack. In the text domain, replacing several words using word-level attack such as Textfooler [85] can lead to a result where the meaning of the sentence has changed substantially, or the sentence has been rendered incoherent due to the selection of an “equivalent” word that does not correctly fit the context. Character-level attacks that perform character replacements and swaps eventually begin to damage the fluency and credibility of the resulting text [63]. A phishing email supposedly sent from a bank, but filled with random typos, would likely fool fewer people. As a result of this, adversarial attacks that fool detection algorithms may fail to fulfill their original purpose in terms of propagating the intended disinformation, or in persuading someone to click a malicious link. Observed decreases in MAUVE scores in successful attack text accompany increased adversarial robustness, in past research on detection of machine generated text [40]. Future applied research might incorporate measures of whether adversarial text that bypassing detection systems would still be effective against targets.

5.4 Interpretability and Fairness of Detection Methods

In the event that an individual is negatively impacted by a machine generated text detection algorithm, it is important that they have recourse to an explanation as to why a decision is made, and to appeal it if it is erroneous. The requirement to provide a human-understandable explanation of why significant decisions have been made is an important part of trustworthy AI policies, which is reflected in current government regulatory guidelines and technology standards related to automated decision making [55, 56, 135, 169], and has influenced NLG model usage policies [59] (discussed further in §5.7). Early work has been done leveraging random forest models and XGBoost for detection of GPT-2 generated fake reviews, with the goal of providing SHapley Additive exPlanations (SHAP) [118] in machine generated text detection [97]. There is a need for future work on methods that are both effective and explainable for machine generated text detection.

The usage of machine learning models to perform positive detection of machine generated text for the purposes of preventing abuse constitutes a situation where such models are likely to have a negative impact on flagged individuals. These penalties may range from relatively lightweight (e.g., having to perform a CAPTCHA challenge to post a comment) to more severe (e.g., denial of a scholarship, or social media ban). As a result, as with other automated decision making systems, it is important that such systems operate in a way that is sufficiently fair, transparent, and interpretable to demonstrate that their operation does not cause harm to users. Social or technical research considering potential harms of machine generated text detection is important to ensuring developed systems are ethically acceptable.

A related critical consideration is that certain groups of individuals may be more likely to have their text flagged by machine generated text detection algorithms, either due to characteristics of their writing (such as language background), or due to non-malicious use of translation tools. For example, it is possible that a detection system designed to prevent a political influence campaign operated using NLG models, may inadvertently end up disproportionately targeting all political speech by individuals who do not natively speak the language of discussion, as has been documented in past research of non-NLG political influence campaigns [41]. Research that identifies ways to improve detection while maintaining fairness and preventing widespread discrimination is deeply important.

5.5 Detection Methods Incorporating Human Agency

As previously mentioned, it is possible that detection of machine generated text may result in suppression of specific individuals or communities in social media whose language background or topics of interest disproportionately cause them to be identified as a false positive by a detection model. In order to reduce this likelihood, and other ethical harms, the development of machine generated text models that incorporate a human analyst may be of use. GLTR remains the only tool currently available for detection of machine generated text that explicitly incorporates a human analyst to improve detection [66]. While analysis of GLTR has shown that it has weaknesses, it has also demonstrated that machine text that fools humans is also more easily detected by algorithms [79]. As such, continued development of moderation tools and systems that leave an avenue for human agency and oversight – guiding principles for trustworthy AI – is a positive area of future development. Similar work has already been done in the field of online influence operation research more generally, using Transformer embeddings to chart and cluster social media for free-form exploration by a human analyst [42], a similar approach may be worthwhile for machine generated text detection.

5.6 Detection of Abuse Beyond Text Content

While many of the threat models discussed in §3 can make use of machine generated text detection as part of mitigation strategies, additional methods might be used to facilitate detection outside of text classification. Work on social bot detection includes additional signals, such as IP addresses and timing of messages, though signals in this domain are also becoming harder to detect over time [165]. Chatbot detection can incorporate features derived from human responses [22]. Prompt injection may bait social bots into exposing themselves [186].

On social media platform, in addition to such technical approaches, it is likely that many platforms will enact policy approaches to increase verification as well. These may take the form of additional verification of users in order to provide a greater barrier to entry for fraudulent accounts. Increased CAPTCHA challenges are already commonplace when platforms are accessed via IP addresses, or accounts are registered with phone numbers associated with a voice-over-IP (VOIP) service [72]. These types of restrictions may become more stringent, with increased user vetting by checking selectors (IP addresses, emails) with third-party reputation services. While the extent of these measures will vary by platform, it is possible that certain platforms may resort to more stringent verification of an individual's real identity using national IDs. In any case, the asymmetric difficulty of defense versus attack in the current threat environment means that increased scrutiny of new accounts will likely be required to avoid a collapse of trust in online spaces.

5.7 Defining Model Usage and Disclosure Policies

Undisclosed usage of AI-generated text content is likely to continue to increase, particularly as NLG models are deployed in user-friendly tools such as Jasper [82] designed to assist with producing articles and social media content. Increased usage of such tools for generating targeted content may result in situations where individuals online are interacting heavily with content predominantly generated by AI models.

This is cause for concern not just due to the erosion of trustworthy AI principles by not disclosing usage of AI models to humans interacting with the content [56], but also of additional ethical concern as NLG models have been found to magnify algorithmic biases found in the content they were trained upon [163]. Digital content farms may begin publishing large amounts of predominantly AI-generated text content (articles, blogs, posts, tweets, etc.) and targeting this content towards the audience most likely to engage with it. Without oversight, this would include highly optimized content that caters to an audience's worst biases and fears – likely a profitable strategy, as anger and anxiety have a strong link with online virality [19]. Moderation strategies for AI-generated content may include limitations to its use, or notifying readers that they are engaging with AI-generated content to allow them to reconsider how much trust they place in what they are reading.

Usage and disclosure policies for online platforms are a worthwhile area of future development, whether those take the role of bans (such as those seen related to generative art [54]), or enforced rules that mandate public disclosure of AI-generated content. Researchers can also take steps themselves by adjusting the licenses of released models to mandate disclosure. AI model BLOOM was released under the first version of the Responsible AI License (RAIL) [59]. The conditions of this license include a requirement for disclosure, an explicit ban on malicious abuse, and a prohibition of specific use-cases (including automated decision making with a potential negative impact, which aligns with terminology under EU [55] and Canadian regulation [169]). Adoption of such a license where appropriate, or development of similar licenses, is an important area of consideration to improve best practices around handling powerful NLG models.

6 CONCLUSION

In this survey, we provided a comprehensive overview of detection methods for machine generated text, carefully evaluating the technical and social benefits of different approaches and including novel research focusing on topics such as adversarial robustness and explainability. We provided context for this review with an overview of natural language generation (NLG) models, and a deep analysis of current threat models. Our exploration of threat models, when viewed alongside our survey on applied detection research, suggests that current domain-specific defenses are not adequate to defend against the vast majority of upcoming threat models.

Our central conclusion is that the field of machine generated text detection has a multitude of open problems that urgently need attention in order to provide suitable defenses against widely-available NLG models. Existing detection methodologies often do not reflect realistic settings of class imbalance and unknown generative model parameters/architectures, nor do they incorporate sufficient transparency and fairness methods to ensure that such detection systems will not themselves cause harm. Preventing widespread abuse of NLG models will require coordinated effort across technical and social domains — alignment between AI researchers, cybersecurity professionals, and non-technical experts will be essential for humans to realize the benefits of high-capacity NLG systems while reducing the societal damage caused by their inevitable abuse.

REFERENCES

- [1] Saeed Abu-Nimeh, Dario Nappa, et al. 2007. A comparison of machine learning techniques for phishing detection. In *Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit*. 60–69.
- [2] David Ifeoluwa Adelani, Haotian Mai, et al. 2020. Generating Sentiment-Preserving Fake Online Reviews Using Neural Language Models and Their Human- and Machine-Based Detection. In *Advanced Information Networking and Applications*, Leonard Barolli, Flora Amato, et al. (Eds.). Springer International Publishing, Cham, 1341–1354.
- [3] Armen Aghajanyan, Akshat Shrivastava, et al. 2021. Better Fine-Tuning by Reducing Representational Collapse. In *ICLR*.
- [4] Luis von Ahn, Manuel Blum, et al. 2003. CAPTCHA: Using hard AI problems for security. In *International conference on the theory and applications of cryptographic techniques*. Springer, 294–311.
- [5] Zainab Alkhalil, Chaminda Hewage, et al. 2021. Phishing Attacks: A Recent Comprehensive Study and a New Anatomy. *Frontiers in Computer Science* 3 (2021). <https://doi.org/10.3389/fcomp.2021.563060>
- [6] American Press Institute. 2014. How Americans get their news. <https://www.americanpressinstitute.org/publications/reports/survey-research/how-americans-get-news/>
- [7] Phoenix CS Andrews. 2021. What is Brigading? <https://institute.global/policy/social-media-futures-what-brigading>
- [8] Wissam Antoun, Fady Baly, et al. 2020. AraBERT: Transformer-based Model for Arabic Language Understanding. *ArXiv* (2020).
- [9] Brooke Auxier and Monica Anderson. 2021. Social media use in 2021. <https://www.pewresearch.org/internet/2021/04/07/social-media-use-in-2021/>
- [10] Alexei Baevski, Wei-Ning Hsu, et al. 2021. Unsupervised speech recognition. *NeurIPS* 34 (2021), 27826–27839.
- [11] Anton Bakhtin, Yuntian Deng, et al. 2021. Residual Energy-Based Models for Text. *J. of Mach. Learn. Res.* 22 (2021), 40–1.
- [12] Anton Bakhtin, Sam Gross, et al. 2019. Real or Fake? Learning to Discriminate Machine from Human Generated Text. *CoRR* abs/1906.03351 (2019). arXiv:1906.03351 <http://arxiv.org/abs/1906.03351>
- [13] Shahryar Baki, Rakesh Verma, et al. 2017. Scaling and Effectiveness of Email Masquerade Attacks: Exploiting Natural Language Generation. In *Proc. 2017 ACM ASIA CCS '17*. ACM, New York, NY, USA, 469–482. <https://doi.org/10.1145/3052973.3053037>

- [14] Nathalie Baracaldo, Bryant Chen, et al. 2017. Mitigating poisoning attacks on machine learning models: A data provenance based approach. In *Proc. 10th ACM Workshop on Artificial Intelligence and Security*. 103–110.
- [15] Leonard E Baum and Ted Petrie. 1966. Statistical inference for probabilistic functions of finite state Markov chains. *The annals of mathematical statistics* 37, 6 (1966), 1554–1563.
- [16] Jason Baumgartner, Savvas Zannettou, et al. 2020. The Pushshift Reddit Dataset. *CoRR* abs/2001.08435 (2020). arXiv:2001.08435 <https://arxiv.org/abs/2001.08435>
- [17] Colin Bellinger, Shiven Sharma, et al. 2012. One-Class versus Binary Classification: Which and When?. In *2012 11th International Conference on Machine Learning and Applications*, Vol. 2. 102–106. <https://doi.org/10.1109/ICMLA.2012.212>
- [18] Daria Beresneva. 2016. Computer-generated text detection using machine learning: A systematic review. In *International Conference on Applications of Natural Language to Information Systems*. Springer, 421–426.
- [19] Jonah Berger and Katherine L Milkman. 2012. What makes online content viral? *Journal of marketing research* 49, 2 (2012), 192–205.
- [20] Mathias Berglund, Tapani Raiko, et al. 2015. Bidirectional Recurrent Neural Networks as Generative Models. In *NIPS*.
- [21] Meghana Moorthy Bhat and Srinivasan Parthasarathy. 2020. How Effectively Can Machines Defend Against Machine-Generated Fake News? An Empirical Study. In *Proc. First Workshop on Insights from Negative Results in NLP*. ACL, Online, 48–53. <https://doi.org/10.18653/v1/2020.insights-1.7>
- [22] Paras Bhatt and Anthony Rios. 2021. Detecting Bot-Generated Text by Characterizing Linguistic Accommodation in Human-Bot Interactions. *CoRR* abs/2106.01170 (2021). arXiv:2106.01170 <https://arxiv.org/abs/2106.01170>
- [23] Stella Biderman and Edward Raff. 2022. Fooling moss detection with pretrained language models. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 2933–2943.
- [24] Ron Bitton, Nadav Maman, et al. 2021. A Framework for Evaluating the Cybersecurity Risk of Real World, Machine Learning Production Systems. *CoRR* abs/2107.01806 (2021). arXiv:2107.01806 <https://arxiv.org/abs/2107.01806>
- [25] Sid Black, Stella Biderman, et al. 2022. GPT-NeoX-20B: An Open-Source Autoregressive Language Model. arXiv:cs.CL/2204.06745
- [26] Zoe Braiterman, Adam Shostack, et al. 2020. <https://www.threatmodelingmanifesto.org/>
- [27] CSET Policy Brief. 2021. AI and the Future of Disinformation Campaigns. (2021).
- [28] Siri Bromander, Audun Jøsang, et al. 2016. Semantic Cyberthreat Modelling. In *STIDS*. 74–78.
- [29] Tom B. Brown, Benjamin Mann, et al. 2020. Language Models are Few-Shot Learners. *ArXiv* abs/2005.14165 (2020).
- [30] A. J. Burns, M. Eric Johnson, et al. 2019. Spear phishing in a barrel: Insights from a targeted phishing campaign. *J. Organ. Comput. Electron. Commer.* 29, 1 (2019), 24–39. <https://doi.org/10.1080/10919392.2019.1552745> arXiv:<https://doi.org/10.1080/10919392.2019.1552745>
- [31] Guillaume Cabanac and Cyril Labbé. 2021. Prevalence of nonsensical algorithmically generated papers in the scientific literature. *Journal of the Association for Information Science and Technology* 72, 12 (2021), 1461–1476.
- [32] Nicholas Carlini, Florian Tramèr, et al. 2020. Extracting Training Data from Large Language Models. *CoRR* abs/2012.07805 (2020). arXiv:2012.07805 <https://arxiv.org/abs/2012.07805>
- [33] Anirban Chakraborty, Manaar Alam, et al. 2018. Adversarial attacks and defences: A survey. *arXiv preprint arXiv:1810.00069* (2018).
- [34] Mark Chen, Jerry Tworek, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374* (2021).
- [35] Xingyuan Chen, Peng Jin, et al. 2022. Automatic Detection of Chinese Generated Essayss Based on Pre-trained BERT. In *2022 IEEE 10th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, Vol. 10. IEEE, 2257–2260.
- [36] Kang Leng Chiew, Kelvin Sheng Chek Yong, et al. 2018. A survey of phishing attacks: Their types, vectors and technical approaches. *Expert Systems with Applications* 106 (2018), 1–20. <https://doi.org/10.1016/j.eswa.2018.03.050>
- [37] Jan K Chorowski, Dzmitry Bahdanau, et al. 2015. Attention-based models for speech recognition. *NeurIPS* 28 (2015).
- [38] Elizabeth Clark, Tal August, et al. 2021. All That’s ‘Human’ Is Not Gold: Evaluating Human Evaluation of Generated Text. In *Proc. 59th Annual Meeting of the ACL and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. ACL, Online, 7282–7296. <https://doi.org/10.18653/v1/2021.acl-long.565>
- [39] Jordan Clive, Kris Cao, et al. 2021. Control prefixes for text generation. *arXiv preprint arXiv:2110.08329* (2021).
- [40] Evan Crothers, Nathalie Japkowicz, et al. 2022. Adversarial Robustness of Neural-Statistical Features in Detection of Generative Transformers. *arXiv preprint arXiv:2203.07983* (2022).
- [41] Evan Crothers, Nathalie Japkowicz, et al. 2019. Towards ethical content-based detection of online influence campaigns. In *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 1–6.
- [42] Evan Crothers, Herna Viktor, et al. 2021. Mean User-Text Agglomeration (MUTA): Practical User Representation and Visualization for Detection of Online Influence Operations. In *International Conference on Computational Data and Social Networks*. Springer, 305–318.
- [43] Yiming Cui, Wanxiang Che, et al. 2021. Pre-training with whole word masking for chinese bert. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 29 (2021), 3504–3514.
- [44] Joseph Cutler, Liam Dugan, et al. 2022. Automatic Detection of Hybrid Human-Machine Text Boundaries. (2022). <https://www.cis.upenn.edu/~jwc/assets/nlp.pdf>
- [45] Slawomir Dadas, Michal Perelkiewicz, et al. 2020. Pre-training polish transformer-based language models at scale. In *International Conference on Artificial Intelligence and Soft Computing*. Springer, 301–314.

- [46] Sumanth Dathathri, Andrea Madotto, et al. 2020. Plug and Play Language Models: A Simple Approach to Controlled Text Generation. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=H1edEyBKDS>
- [47] Nassim Dehouche. 2021. Plagiarism in the age of massive Generative Pre-trained Transformers (GPT-3). *Ethics in Science and Environmental Politics* 21 (2021), 17–23.
- [48] Nina Dethlefs and Heriberto Cuayáhuatl. 2010. Hierarchical Reinforcement Learning for Adaptive Text Generation. In *Proc. 6th INLG. ACL*. <https://aclanthology.org/W10-4204>
- [49] Jacob Devlin, Ming-Wei Chang, et al. 2018. BERT: Pre-training of Deep Bidirectional Transformers. *CoRR* abs/1810.04805 (2018). arXiv:1810.04805 <http://arxiv.org/abs/1810.04805>
- [50] Chenhe Dong, Yinghui Li, et al. 2022. A Survey of Natural Language Generation. *ACM Comput. Surv.* (jul 2022). <https://doi.org/10.1145/3554727> Just Accepted.
- [51] Yao Dou, Maxwell Forbes, et al. 2022. Is GPT-3 Text Indistinguishable from Human Text? Scarecrow: A Framework for Scrutinizing Machine Text. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 7250–7274.
- [52] Pablo Ariel Duboue and Kathleen R. McKeown. 2003. Statistical Acquisition of Content Selection Rules for Natural Language Generation. In *Proc. 2003 Conference on Empirical Methods in Natural Language Processing*. 121–128. <https://aclanthology.org/W03-1016>
- [53] Liam Dugan, Daphne Ippolito, et al. 2020. RoFT: A Tool for Evaluating Human Detection of Machine-Generated Text. *ArXiv* abs/2010.03070 (2020).
- [54] Benj Edwards. 2022. Flooded with AI-generated images, some art communities ban them completely. <https://arstechnica.com/information-technology/2022/09/flooded-with-ai-generated-images-some-art-communities-ban-them-completely/>
- [55] European Commission. 2018. Guidelines on Automated individual decision-making and Profiling for the purposes of Regulation. <https://ec.europa.eu/newsroom/article29/items/612053/en>
- [56] European Commission and Directorate-General for Communications Networks, Content and Technology. 2019. *Ethics guidelines for trustworthy AI*. Publications Office. <https://doi.org/doi/10.2759/346720>
- [57] Tiziano Fagni, Fabrizio Falchi, et al. 2021. TweepFake: About detecting deepfake tweets. *PLOS ONE* 16, 5 (05 2021), 1–16. <https://doi.org/10.1371/journal.pone.0251415>
- [58] Xiaocheng Feng, Ming Liu, et al. 2018. Topic-to-essay generation with neural networks.. In *IJCAI*. 4078–4084.
- [59] Carlos Muñoz Ferrandis, Danish Contractor, et al. 2022. BigScience RAIL License v1.0. <https://huggingface.co/spaces/bigscience/license>
- [60] Leon Fröhlhing and Arkaitz Zubiaga. 2021. Feature-based detection of automated language models: tackling GPT-2, GPT-3 and Grover. *PeerJ Computer Science* 7 (2021), e443.
- [61] Maksym Gabielkov, Arthi Ramachandran, et al. 2016. Social Clicks: What and Who Gets Read on Twitter?. In *ACM SIGMETRICS / IFIP Performance 2016*. Antibes Juan-les-Pins, France. <https://hal.inria.fr/hal-01281190>
- [62] Matthias Gallé, Jos Rozen, et al. 2021. Unsupervised and Distributional Detection of Machine-Generated Text. arXiv:cs.CL/2111.02878
- [63] Ji Gao, Jack Lanchantin, et al. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, 50–56.
- [64] Leo Gao, Stella Biderman, et al. 2021. The Pile: An 800GB Dataset of Diverse Text for Language Modeling. *CoRR* abs/2101.00027 (2021). arXiv:2101.00027 <https://arxiv.org/abs/2101.00027>
- [65] Albert Gatt and Emiel Kraemer. 2018. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research* 61 (2018), 65–170.
- [66] Sebastian Gehrmann, Hendrik Strobelt, et al. 2019. GLTR: Statistical Detection and Visualization of Generated Text. In *Proc. 57th Annual Meeting of the ACL: System Demonstrations*. ACL, Florence, Italy, 111–116. <https://doi.org/10.18653/v1/P19-3019>
- [67] Alberto Giaretta and Nicola Dragoni. 2020. Community Targeted Phishing. In *Proceedings of 6th International Conference in Software Engineering for Defence Applications*, Paolo Ciancarini, Manuel Mazzara, et al. (Eds.). Springer International Publishing, Cham, 86–93.
- [68] Drew Gooden. 2022. using AI to write a youtube video. Retrieved September 4, 2022 from <https://www.youtube.com/watch?v=BaVpeJlcQzg>
- [69] Ian Goodfellow, Jean Pouget-Abadie, et al. 2020. Generative adversarial networks. *Commun. ACM* 63, 11 (2020), 139–144.
- [70] Riley Goodside. 2022. Exploiting GPT-3 prompts with malicious inputs that order the model to ignore its previous directions. <https://twitter.com/goodside/status/1569128808308957185>
- [71] Sergio Guadarrama, Niveda Krishnamoorthy, et al. 2013. YouTube2Text: Recognizing and Describing Arbitrary Activities Using Semantic Hierarchies and Zero-Shot Recognition. In *2013 IEEE ICCV*. 2712–2719. <https://doi.org/10.1109/ICCV.2013.337>
- [72] Meriem Guerar, Luca Verderame, et al. 2021. Gotta CAPTCHA 'Em All: A Survey of 20 Years of the Human-or-Computer Dilemma. *ACM Comput. Surv.* 54, 9, Article 192 (oct 2021), 33 pages. <https://doi.org/10.1145/3477142>
- [73] Jeffrey Hargrave. 2005. SCiGen - An Automatic CS Paper Generator. <https://pdos.csail.mit.edu/archive/scigen/>
- [74] Hamza Harkous, Isabel Groves, et al. 2020. Have your text and use it too! End-to-end neural data-to-text generation with semantic fidelity. In *COLING 2020*. <https://www.amazon.science/publications/have-your-text-and-use-it-too-end-to-end-neural-data-to-text-generation-with-semantic-fidelity>

- [75] Sherry He, Brett Hollenbeck, et al. 2022. The market for fake reviews. *Marketing Science* (2022).
- [76] Ari Holtzman, Jan Buys, et al. 2019. The Curious Case of Neural Text Degeneration. *CoRR* abs/1904.09751 (2019). arXiv:1904.09751 <http://arxiv.org/abs/1904.09751>
- [77] Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146* (2018).
- [78] Aminul Huq and Mst. Tasnim Pervin. 2020. Adversarial Attacks and Defense on Texts: A Survey. *CoRR* abs/2005.14108 (2020). arXiv:2005.14108 <https://arxiv.org/abs/2005.14108>
- [79] Daphne Ippolito, Daniel Duckworth, et al. 2020. Automatic Detection of Generated Text is Easiest when Humans are Fooled. In *ACL*.
- [80] Srinivasan Janarthanam and Oliver Lemon. 2009. Learning lexical alignment policies for generating referring expressions for spoken dialogue systems. In *Proc. 12th ENLG*. 74–81.
- [81] Nathalie Japkowicz et al. 2000. Learning from imbalanced data sets: a comparison of various strategies. In *AAAI workshop on learning from imbalanced data sets*, Vol. 68. AAAI Press Menlo Park, CA, 10–15.
- [82] Jasper AI. 2022. The best ai writing assistant. Retrieved September 4, 2022 from <https://www.jasper.ai/>
- [83] Ganesh Jawahar, Muhammad Abdul-Mageed, et al. 2020. Automatic Detection of Machine Generated Text: A Critical Survey. In *Proc. 28th International Conference on Computational Linguistics*. 2296–2309.
- [84] Di Jin, Zhijing Jin, et al. 2022. Deep Learning for Text Style Transfer: A Survey. *Computational Linguistics* 48, 1 (March 2022), 155–205. https://doi.org/10.1162/coli_a_00426
- [85] Di Jin, Zhijing Jin, et al. 2019. Is BERT Really Robust? Natural Language Attack on Text Classification and Entailment. *CoRR* abs/1907.11932 (2019). arXiv:1907.11932 <http://arxiv.org/abs/1907.11932>
- [86] Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proc. 2013 conference on empirical methods in natural language processing*. 1700–1709.
- [87] Ambedkar Kanapala, Sukomal Pal, et al. 2019. Text summarization from legal documents: a survey. *Artificial Intelligence Review* 51, 3 (2019), 371–402.
- [88] Davinder Kaur, Suleyman Uslu, et al. 2022. Trustworthy Artificial Intelligence: A Review. *ACM Comput. Surv.* 55, 2, Article 39 (jan 2022), 38 pages. <https://doi.org/10.1145/3491209>
- [89] Simon Kemp. 2022. Content across cultures. <https://datareportal.com/reports/future-trends-2022-cross-cultural-content>
- [90] Simon Kemp. 2022. Digital 2022 Global Digital Overview. <https://datareportal.com/reports/digital-2022-global-overview-report>
- [91] Nitish Shirish Keskar, Bryan McCann, et al. 2019. CTRL: A Conditional Transformer Language Model for Controllable Generation. *ArXiv* abs/1909.05858 (2019).
- [92] Michael Khurshchev, Ruslan Vasilev, et al. 2022. YaLM 100B. "<https://huggingface.co/yandex/yalm-100b>".
- [93] Yannic Kilcher. 2022. This is the worst ai ever. <https://www.youtube.com/watch?v=efPrteLdcdM>
- [94] Soomin Kim, Joonhwan Lee, et al. 2019. Comparing data from chatbot and web surveys: Effects of platform and conversational style on survey response quality. In *Proc. 2019 CHI conference on human factors in computing systems*. 1–12.
- [95] Loren Kohnfelder and Praerit Garg. 1999. The threats to our products. *Microsoft Interface, Microsoft Corporation* 33 (1999).
- [96] Ravikumar Kondadadi, Blake Howald, et al. 2013. A statistical nlg framework for aggregated planning and realization. In *Proc. 51st Annual Meeting of the ACL (Volume 1: Long Papers)*. 1406–1415.
- [97] Peter Kowalczyk, Marco Röder, et al. 2022. Detecting and Understanding Textual Deepfakes in Online Reviews. (2022).
- [98] Ben Krause, Akhilesh Deepak Gotmare, et al. 2020. Gedi: Generative discriminator guided sequence generation. *arXiv preprint arXiv:2009.06367* (2020).
- [99] Andrey Kurenkov. 2022. Lessons from the GPT-4Chan Controversy. <https://thegradient.pub/gpt-4chan-lessons>. *The Gradient* (2022).
- [100] Laida Kushnareva, Daniil Cherniavskii, et al. 2021. Artificial Text Detection via Examining the Topology of Attention Maps. *CoRR* abs/2109.04825 (2021). arXiv:2109.04825 <https://arxiv.org/abs/2109.04825>
- [101] Cyril Labbé and Dominique Labbé. 2013. Duplicate and fake publications in the scientific literature: how many SCIdgen papers in computer science? *Scientometrics* 94, 1 (2013), 379–396.
- [102] Irene Langkilde and Kevin Knight. 1998. Generation that exploits corpus-based statistical knowledge. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*.
- [103] Majd Latah. 2020. Detection of malicious social bots: A survey and a refined taxonomy. *Expert Systems with Applications* 151 (2020), 113383. <https://doi.org/10.1016/j.eswa.2020.113383>
- [104] Allen Lavoie and Mukkai S. Krishnamoorthy. 2010. Algorithmic Detection of Computer Generated Text. *ArXiv* abs/1008.0706 (2010).
- [105] Yann LeCun, Sumit Chopra, et al. 2006. A tutorial on energy-based learning. *Predicting structured data* 1, 0 (2006).
- [106] Mike Lewis, Yinhan Liu, et al. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proc. 58th Annu. Meet. ACL Sys. Demo*. ACL, Online, 7871–7880. <https://doi.org/10.18653/v1/2020.acl-main.703>
- [107] Guang Li, Linchao Zhu, et al. 2019. Entangled transformer for image captioning. In *Proc. IEEE/CVF*. 8928–8937.
- [108] Haoran Li, Junnan Zhu, et al. 2017. Multi-modal Summarization for Asynchronous Collection of Text, Image, Audio and Video. In *Proc. 2017 Conf. EMNLP, ACL, Copenhagen, Denmark*, 1092–1102. <https://doi.org/10.18653/v1/D17-1114>

- [109] Jiwei Li, Will Monroe, et al. 2016. Deep Reinforcement Learning for Dialogue Generation. In *EMNLP*.
- [110] Junyi Li, Tianyi Tang, et al. 2021. Pretrained Language Model for Text Generation: A Survey. In *Proc. 30th IJCAI*, Zhi-Hua Zhou (Ed.). IJCAI Organization, 4492–4499. <https://doi.org/10.24963/ijcai.2021/612> Survey Track.
- [111] Ziming Li, Julia Kiseleva, et al. 2018. Dialogue Generation: From Imitation Learning to Inverse Reinforcement Learning. *CoRR* abs/1812.03509 (2018). arXiv:1812.03509 <http://arxiv.org/abs/1812.03509>
- [112] Percy Liang, Rishi Bommasani, et al. 2022. The Time Is Now to Develop Community Norms for the Release of Foundation Models. <https://crfm.stanford.edu/2022/05/17/community-norms.html>
- [113] Percy Liang and Rob Reich. 2022. Condemning the deployment of gpt-4chan. <https://docs.google.com/forms/d/e/1FAIpQLSdh3Pgh0sGrYtRihBu-GPN7FSQoODBLvF7dVAFLZk2iuMgoLw/viewform>
- [114] Kevin Lin, Dianqi Li, et al. 2017. Adversarial Ranking for Language Generation. In *NIPS*.
- [115] Yinhan Liu, Myle Ott, et al. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *ArXiv* abs/1907.11692 (2019).
- [116] Bryan Lufkin. 2021. Why do cover letters still exist? <https://www.bbc.com/worklife/article/20211005-why-do-cover-letters-still-exist>
- [117] Hans Peter Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of research and development* 2, 2 (1958), 159–165.
- [118] Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *Advances in neural information processing systems* 30 (2017).
- [119] John Lyons. 1991. *Natural Language and Universal Grammar: Volume 1: Essays in Linguistic Theory*. Vol. 1. Cambridge University Press.
- [120] Arya Manjaramkar. 2021. CodeGenX. <https://github.com/DeepGenX/CodeGenX>.
- [121] Louis Martin, Benjamin Muller, et al. 2019. CamemBERT: a Tasty French Language Model. *ArXiv* abs/1911.03894 (2019).
- [122] Kirill Meleshevich and Bret Schafer. 2018. Online information laundering: The role of social media. <https://securingdemocracy.gmfus.org/online-information-laundering-the-role-of-social-media/>
- [123] Stephen Merity, Nitish Shirish Keskar, et al. 2017. Regularizing and optim. LSTM lang. models. *arXiv preprint arXiv:1708.02182* (2017).
- [124] Albert Meroño-Peñuela and Dayana Spagnuolo. 2020. Can a Transformer Assist in Scientific Writing? Generating Semantic Web Paper Snippets with GPT-2. In *The Semantic Web: ESWC 2020 Satellite Events*, Andreas Harth, Valentina Presutti, et al. (Eds.). Springer International Publishing, Cham, 158–163.
- [125] Tomáš Mikolov et al. 2012. Stat. lang. models based on neural networks. *Presentation at Google, Mountain View, 2nd April* 80, 26 (2012).
- [126] Jaron Mink, Licheng Luo, et al. 2022. DeepPhish: Understanding User Trust Towards Artificially Generated Profiles in Online Social Networks. In *31st USENIX Security Symposium (USENIX Security 22)*. USENIX Association, Boston, MA, 1669–1686. <https://www.usenix.org/conference/usenixsecurity22/presentation/mink>
- [127] Shaoor Munir, Brishna Batool, et al. 2021. Through the Looking Glass: Learning to Attribute Synthetic Text Generated by Language Models. In *Proc. 16th Conf. Euro. Chap. ACL: Main Vol. ACL*, Online, 1811–1822. <https://doi.org/10.18653/v1/2021.eacl-main.155>
- [128] Hoang-Quoc Nguyen-Son, Ngoc-Dung T. Tieu, et al. 2017. Identifying computer-generated text using statistical analysis. In *2017 APSIPA ASC*. 1504–1511. <https://doi.org/10.1109/APSIPA.2017.8282270>
- [129] Open Cover Letter. 2022. Open Cover Letter: Generate Cover Letters with AI. <https://www.opencoverletter.com/>
- [130] OpenAI. 2020. GPT-3. <https://github.com/openai/gpt-3>.
- [131] Myle Ott, Sergey Edunov, et al. 2019. fairseq: A Fast, Extensible Toolkit for Sequence Modeling. In *Proc. of NAACL-HLT 2019: Demo*.
- [132] Papers With Code. 2022. Natural language processing benchmarks. <https://paperswithcode.com/area/natural-language-processing>
- [133] Dipti Pawade, Avani M. Sakhapara, et al. 2018. Story Scrambler - Automatic Text Generation Using Word Level RNN-LSTM. *International Journal of Information Technology and Computer Science* (2018).
- [134] Rivindu Perera and Parma Nand. 2017. Recent advances in natural language generation: A survey and classification of the empirical literature. *Computing and Informatics* 36, 1 (2017), 1–32.
- [135] P Jonathon Phillips, Carina A Hahn, et al. 2020. Four principles of explainable artificial intelligence. *Gaithersburg, Maryland* (2020).
- [136] P Jonathon Phillips, Amy N Yates, et al. 2018. Face recognition accuracy of forensic examiners, superrecognizers, and face recognition algorithms. *Proc. National Academy of Sciences* 115, 24 (2018), 6171–6176.
- [137] Alec Radford, Jeff Wu, et al. 2019. Language Models are Unsupervised Multitask Learners. (2019).
- [138] Alec Radford, Jeff Wu, et al. 2019. GPT-2 Output Dataset. <https://github.com/openai/gpt-2-output-dataset>.
- [139] Colin Raffel, Noam Shazeer, et al. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *J. of Mach. Learn. Res.* 21, 140 (2020), 1–67. <http://jmlr.org/papers/v21/20-074.html>
- [140] Aditya Ramesh, Mikhail Pavlov, et al. 2021. Zero-shot text-to-image generation. In *ICML*. PMLR, 8821–8831.
- [141] Priyanka Ranade, Aritrnan Piplai, et al. 2021. Generating Fake Cyber Threat Intelligence Using Transformer-Based Models. In *2021 International Joint Conference on Neural Networks (IJCNN)*. 1–9. <https://doi.org/10.1109/IJCNN52387.2021.9534192>
- [142] Reddit. 2018. Reddit Transparency Report: Suspicious Accounts. <https://www.reddit.com/wiki/suspiciousaccounts>.
- [143] Scott Reed, Konrad Zolna, et al. 2022. A Generalist Agent. <https://doi.org/10.48550/ARXIV.2205.06175>
- [144] Ehud Reiter and Robert Dale. 2002. Building Applied Natural Language Generation Systems. *Natural Language Engineering* 3 (03 2002). <https://doi.org/10.1017/S1351324997001502>

- [145] Juan Rodriguez, Todd Hay, et al. 2022. Cross-Domain Detection of GPT-2-Generated Technical Text. In *Proc. 2022 Conf. Nor. Amer. Chapt. ACL: Human Language Technologies*. ACL, Seattle, United States, 1213–1233. <https://doi.org/10.18653/v1/2022.naacl-main.88>
- [146] Robin Rombach, Andreas Blattmann, et al. 2021. High-Resolution Image Synthesis with Latent Diffusion Models. arXiv:cs.CV/2112.10752
- [147] Domenic Rosati. 2022. SynSciPass: detecting appropriate uses of scientific text generation. <https://doi.org/10.48550/ARXIV.2209.03742>
- [148] Joni Salminen, Chandrashekhar Kandpal, et al. 2022. Creating and detecting fake reviews of online products. *Journal of Retailing and Consumer Services* 64 (2022), 102771. <https://doi.org/10.1016/j.jretconser.2021.102771>
- [149] Sashank Santhanam and Samira Shaikh. 2019. A survey of natural language generation techniques with a focus on dialogue systems-past, present and future directions. *arXiv preprint arXiv:1906.00500* (2019).
- [150] Teven Le Scao, Angela Fan, et al. 2022. BLOOM: A 176B-Parameter Open-Access Multilingual Language Model. <https://doi.org/10.48550/ARXIV.2211.05100>
- [151] Elliot Schrage and David Ginsberg. 2018. Facebook Launches New Initiative to Help Scholars Assess Social Media’s Impact on Elections. <https://newsroom.fb.com/news/2018/04/new-elections-initiative/>
- [152] Roi Schuster, Congzheng Song, et al. 2021. You autocomplete me: Poisoning vulnerabilities in neural code completion. In *30th USENIX Security Symposium (USENIX Security 21)*. 1559–1575.
- [153] Abigail See, Aneesh Pappu, et al. 2019. Do Massively Pretrained Language Models Make Better Storytellers?. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. 843–861.
- [154] Andrew D. Selbst, Danah Boyd, et al. 2019. Fairness and Abstraction in Sociotechnical Systems. In *Proc. Conference on Fairness, Accountability, and Transparency (FAT* ’19)*. ACM, New York, NY, USA, 59–68. <https://doi.org/10.1145/3287560.3287598>
- [155] Tatiana Shamardina, Vladislav Mikhailov, et al. 2022. Findings of the The RuATD Shared Task 2022 on Artificial Text Detection in Russian. *arXiv preprint arXiv:2206.01583* (2022).
- [156] Fadi Abu Sheikha and Diana Inkpen. 2011. Generation of formal and informal sentences. In *Proc. 13th European Workshop on Natural Language Generation*. 187–193.
- [157] Zhan Shi, Xinchu Chen, et al. 2018. Toward Diverse Text Generation with Inverse Reinforcement Learning. In *IJCAI*.
- [158] Adam Shostack. 2014. *Threat modeling: Designing for security*. John Wiley & Sons.
- [159] Adam Shostack. 2021. Shostack’s 4 Question Frame for Threat Modeling. <https://github.com/adamshostack/4QuestionFrame>.
- [160] Kai Shu, Suhang Wang, et al. 2020. Mining disinformation and fake news: Concepts, methods, and recent advancements. In *Disinformation, misinformation, and fake news in social media*. Springer, 1–19.
- [161] Kurt Shuster, Jing Xu, et al. 2022. BlenderBot 3: a deployed conversational agent that continually learns to responsibly engage. <https://doi.org/10.48550/ARXIV.2208.03188>
- [162] SS Skrylnikov, PA Posokhov, et al. 2022. Artificial text detection in Russian language: a BERT-based Approach. In *Dialogue*. <https://doi.org/10.28995/2075-7182-2022-21-470-476>
- [163] Irene Solaiman, Miles Brundage, et al. 2019. Release strategies and the social impacts of language models. *arXiv preprint arXiv:1908.09203* (2019).
- [164] Kaitao Song, Xu Tan, et al. 2019. MASS: Masked Sequence to Sequence Pre-training for Language Generation. In *ICML*. 5926–5936.
- [165] Stefan Stieglitz, Florian Brachten, et al. 2017. Do social bots (still) act different to humans?—Comparing metrics of social bots with those of humans. In *International conference on social computing and social media*. Springer, 379–395.
- [166] Harald Stiff and Fredrik Johansson. 2022. Detecting computer-generated disinformation. *Int. J. Data Sci. Anal.* 13, 4 (2022), 363–383.
- [167] Akhilesh Sudhakar, Bhargav Upadhyay, et al. 2019. “Transforming” Delete, Retrieve, Generate Approach for Controlled Text Style Transfer. In *Proc. 2019 Conf. EMNLP-IJCNLP*. 3269–3279.
- [168] Yi Tay, Dara Bahri, et al. 2020. Reverse Engineering Configurations of Neural Text Generation Models. In *Proc. 58th Annu. Meet. ACL Sys. Demo*. 275–279.
- [169] TB Canada Secretariat. 2021. Directive on automated decision-making. <https://www.tbs-sct.canada.ca/pol/doc-eng.aspx?id=32592>
- [170] Senait G Tesfagergish, Robertas Damaševičius, et al. 2021. Deep Fake Recognition in Tweets Using Text Augmentation, Word Embeddings and Deep Learning. In *International Conference on Computational Science and Its Applications*. Springer, 523–538.
- [171] M. Onat Topal, Anil Bas, et al. 2021. Exploring Transformers in Natural Language Generation: GPT, BERT, and XLNet. *ArXiv abs/2102.08036* (2021).
- [172] Julien Tourille, Babacar Sow, et al. 2022. Automatic Detection of Bot-Generated Tweets. In *Proc. 1st International Workshop on Multimedia AI against Disinformation (MAD ’22)*. ACM, New York, NY, USA, 44–51. <https://doi.org/10.1145/3512732.3533584>
- [173] A. M. Turing. 1950. Computing Machinery and Intelligence. *Mind* LIX, 236 (10 1950), 433–460. <https://doi.org/10.1093/mind/LIX.236.433> arXiv:<https://academic.oup.com/mind/article-pdf/LIX/236/433/30123314/lix-236-433.pdf>
- [174] Twitter. 2019. Twitter Elections Integrity Dataset. https://about.twitter.com/en_us/values/elections-integrity.html. Accessed: 2022-09-23.
- [175] Tony UcedaVelez and Marco M Morana. 2015. *Risk Centric Threat Modeling: process for attack simulation and threat analysis*. John Wiley & Sons.
- [176] Adaku Uchendu, Thai Le, et al. 2020. Authorship Attribution for Neural Text Generation. In *Proc. 2020 Conf. EMNLP*. ACL, Online, 8384–8395. <https://doi.org/10.18653/v1/2020.emnlp-main.673>

- [177] UK National Cyber Security Centre. 2022. Threat modelling. <https://www.ncsc.gov.uk/collection/building-a-security-operations-centre/onboarding-systems-and-log-sources/threat-modelling>
- [178] Ashish Vaswani, Noam M. Shazeer, et al. 2017. Attention is All you Need. *ArXiv abs/1706.03762* (2017).
- [179] Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. <https://github.com/kingoflolz/mesh-transformer-jax>.
- [180] Jun Wang, Mingfei Gao, et al. 2022. TAG: Boosting Text-VQA via Text-aware Visual Question-answer Generation. <https://doi.org/10.48550/ARXIV.2208.01813>
- [181] Jianfeng Wang, Zhengyuan Yang, et al. 2022. GIT: A Generative Image-to-text Transformer for Vision and Language. *arXiv preprint arXiv:2205.14100* (2022).
- [182] Mengqian Wang, Manhua Wang, et al. 2021. A systematic review of automatic text summarization for biomedical literature and EHRs. *Journal of the American Medical Informatics Association* 28, 10 (2021), 2287–2297.
- [183] Wenqi Wang, Benxiao Tang, et al. 2019. A survey on Adversarial Attacks and Defenses in Text. *CoRR abs/1902.07285* (2019). [arXiv:1902.07285](https://arxiv.org/abs/1902.07285) <http://arxiv.org/abs/1902.07285>
- [184] Max Weiss. 2019. Deepfake bot submissions to federal public comment websites cannot be distinguished from human submissions. *Technology Science* (2019).
- [185] Joseph Weizenbaum. 1966. ELIZA—a Computer Program for the Study of Natural Language Communication between Man and Machine. *Commun. ACM* 9, 1 (jan 1966), 36–45. <https://doi.org/10.1145/365153.365168>
- [186] Simon Willison. 2022. <https://simonwillison.net/2022/Sep/12/prompt-injection/>
- [187] Lantao Yu, Weinan Zhang, et al. 2017. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. In *AAAI*.
- [188] Juan Pablo Ossa Zapata. 2022. GPT-3 Prompt Injection Example. <https://twitter.com/mkualquiera/status/1570546998104948736>
- [189] Rowan Zellers. 2019. Why We Released Grover. <https://thegradient.pub/why-we-released-grover/>. *The Gradient* (2019).
- [190] Rowan Zellers, Ari Holtzman, et al. 2019. Defending against neural fake news. *NeurIPS* 32 (2019).
- [191] Wei Zeng, Xiaozhe Ren, et al. 2021. PanGu- α : Large-scale Autoregressive Pretrained Chinese Language Models with Auto-parallel Computation. *CoRR abs/2104.12369* (2021). [arXiv:2104.12369](https://arxiv.org/abs/2104.12369) <https://arxiv.org/abs/2104.12369>
- [192] Li Zhang and Jian-Tao Sun. 2009. *Text Generation*. Springer US, Boston, MA, 3048–3051. https://doi.org/10.1007/978-0-387-39940-9_416
- [193] Susan Zhang, Stephen Roller, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068* (2022).
- [194] Yizhe Zhang, Siqi Sun, et al. 2020. DIALOGPT: Large-Scale Generative Pre-training for Conversational Response Generation. 270–278.
- [195] Luowei Zhou, Hamid Palangi, et al. 2020. Unified vision-language pre-training for image captioning and vqa. In *Proc. AAAI Conference on Artificial Intelligence*, Vol. 34. 13041–13049.
- [196] Ziming Zhuang, Ergin Elmacioglu, et al. 2007. Measuring Conference Quality by Mining Program Committee Characteristics. In *Proc. 7th JCDL (JCDL '07)*. ACM, New York, NY, USA, 225–234. <https://doi.org/10.1145/1255175.1255220>
- [197] George Kingsley Zipf. 1949. Human behavior and the principle of least effort. (1949).