On pushing DeepFake Tweet Detection capabilities to the limits

Margherita Gambini
CNR - Institute of Informatics and Telematics
Pisa, Italy
m.gambini@iit.cnr.it

Tiziano Fagni
CNR - Institute of Informatics and Telematics
Pisa, Italy
t.fagni@iit.cnr.it

Fabrizio Falchi
CNR - Institute of Information Science and Technologies
Pisa, Italy
fabrizio.falchi@cnr.it

Maurizio Tesconi
CNR - Institute of Informatics and Telematics
Pisa, Italy
m.tesconi@iit.cnr.it

ABSTRACT

The recent advances in natural language generation provide an additional tool to manipulate public opinion on social media. Even though there has not been any report of malicious exploit of the newest generative techniques so far, disturbing human-like scholarly examples of GPT-2 and GPT-3 can be found on social media. Therefore, our paper investigates how the state-of-the-art deepfake social media text detectors perform at recognizing GPT-2 tweets as machine-written, also trying to improve the state-of-the-art by hyper-parameter tuning and ensembling the most promising detectors; finally, our work concentrates on studying the detectors' capabilities to generalize over tweets generated by the more sophisticated and complex evolution of GPT-2, that is GPT-3. Results demonstrate that hyper-parameter optimization and ensembling advance the state-of-the-art, especially on the detection of GPT-2 tweets. However, all tested detectors dramatically decreased their accuracy on GPT-3 tweets. Despite this, we found out that even though GPT-3 tweets are much closer to human-written tweets than the ones produced by GPT-2, they still have latent features in common share with other generative techniques like GPT-2, RNN and other older methods. All things considered, the research community should quickly devise methods to detect GPT-3 social media texts, as well as older generative methods.

CCS CONCEPTS

• **Security and privacy → Social aspects of security and privacy**; • **Information systems → Data analytics**.

KEYWORDS

Deepfake Detection, Machine-generated, Twitter, Social Media, GPT-2, GPT-3

ACM Reference Format:
Margherita Gambini, Tiziano Fagni, Fabrizio Falchi, and Maurizio Tesconi. 2022. On pushing DeepFake Tweet Detection capabilities to the limits. In *14th ACM Web Science Conference 2022 (WebSci '22), June 26–29, 2022, Barcelona, Spain.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3501247.3531560

1 INTRODUCTION

Social media offer a faster and deeper way to deliberately spread misinformation with the intention of manipulating and altering people's opinions [37] mainly for economic benefits and to sow distrust in democratic countries. To this aim, accounts of different level of humanness are involved: from sock puppets, trolls and hijacked accounts, to cyborg accounts (human-assisted bots or bot-assisted humans) [6], until completely automated and ever more sophisticated social media accounts (social bots) that try to imitate the human behaviour [9]. In particular, the wide usage of bots [5, 29], combined with the recent advances in natural language generative models, such as the GPT models [7, 25, 26] and Grover [42], empowers the adversaries with a tool to easily spread ever more realistic fake news and messages. Even though the novel Transformer-based language model weren't employed, the Net Neutrality case [17] is emblematic: millions of duplicated and template-based comments might have had a big role in the Federal Communications Commission's decision on the repeal of net neutrality in 2017; this suggests that also cheap text manipulation techniques may sow false believes, hence what could be done with the ever more powerful language models is a major threat to address. Lately, there have been examples of usage of GPT-2 [26] and GPT-3 [7] to probe the USA governmental process [39] and to automatically create blog posts [14] for research purposes, as well as to write tweets to test the generation capabilities [11]. The only case of not declared examples of deepfake texts on the Internet is the one of the *thegentlemetre* bot who posted Reddit comments [15] with no malicious intentions; however, the authors didn't have the rights to use the Philosopher AI [2] based on GPT-3, therefore the bot was taken down after a while. Apart from this scholarly case, there has not been any report of exploiting the newest generative techniques to carry on misinformation operations so far. Nonetheless, it is crucial to stay vigilant and continuously develop autonomous detection systems of machine-generated texts, hereinafter referred to as *deepfakes*, to safeguard true information and democracy on social media.

It is curious that all public deepfake text examples on social media are written by GPT models: maybe it means that the authors behind those texts experimented that the GPT models are the best at writing short social media posts, unlike Grover [43], CTRL [18] or other Transformer-based language models. In any case, [11] showed

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

WebSci '22, June 26–29, 2022, Barcelona, Spain
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9191-7/22/06…$15.00
https://doi.org/10.1145/3501247.3531560

that detecting GPT-2 tweets as machine-generated is much more challenging than recognizing tweets generated by RNN and other older generative techniques (not Transformer-based) as deepfake; on the other hand, they proved that identifying human-written tweets among deepfake tweets is easier, suggesting that GPT-2 can write human-like tweets. Hence, a focus on the detection of GPT-2 social media texts among human-written ones is needed, as well as probing the state-of-the-art detection methods on the generation capabilities of GPT-3 in a social media context.

Starting from the state-of-the-art research about deepfake social media text detection [1, 11, 28, 31, 34], we study whether the detection of GPT-2 social media texts can be improved by optimizing the hyper-parameters of the most promising detection techniques, i.e. Transformer-based language models jointly fine-tuned with a neural network binary classifier over the specific deepfake social media text domain [1, 11, 28]; in particular, we exploit the most popular Transformer-based language models and two language models pre-trained on English tweets. Moreover, we test whether an ensemble of the most promising detectors can increase the performances. Besides, the deepfake datasets are usually balanced over human and machine-generated texts; this doesn't represent a real-setting scenario, where the quantity of machine-generated texts is much less than the human-written ones. Therefore, a different way to evaluate the detectors' performance over a real scenario is provided, that is analysing our detectors' Receiver Operating Characteristic (ROC) curve . Finally, to assess the deepfake text detectors' capabilities on GPT-3 social media texts, the tested detectors are evaluated on a dataset of 3'795 tweets written by eight Twitter bots using GPT-3. Apart from our new set of GPT-3 tweets, the TweepFake dataset is used, since its deepfake tweets are examples of the actual real risk level on social media with respect to machine-generated texts.

Our contributions. The contributions of our work are summarized as follows:

- We studied whether tuning the hyper-parameters during the transfer-learning phase of Transformer-based detectors brings performance improvements or not over the detection of GPT-2 tweets. Results showed that this tuning tends to balance the detection accuracy among the different categories of TweepFake tweets (human-written, GPT2, RNN, Others Techniques), sometimes resulting in a decrease of GPT-2 tweets detection accuracy.
- We investigated the performances of an ensemble stacking learner detector. Results confirmed our hypothesis: ensembling several detection models helps the recognition of GPT-2 tweets as machine-generated, especially in a real-setting scenario.
- We advanced the state-of-the-art on the Deepfake Tweets detection task over the TweepFake dataset through hyper-parameter optimization and ensembling.
- We probed the state-of-the-art deepfake tweet detection methods over GPT-3 tweets, showing a greatly decrease in detection capabilities. This result highlights the lacking of generalization of these detection techniques with respect to more sophisticated and complex generation models like GPT-3 even in a social media context.

The rest of the article is organized as follows: Section 2 reviews previous works on deepfake text detection, Section 3 describes our tested approaches to improve the deepfake tweet detection, whereas Section 4 details the employed dataset. The results are presented and discussed in Section 5.

2 RELATED WORK

Vaswani et al. (2017)'s Transformer architecture [36] set a milestone in the Natural Language Generation (NLG) research field: taking inspiration from it, Transformers-based language models like GPT-2 [26], GPT-3 [7], GROVER [42] and CLTR [18] can autonomously write non-trivial, coherent, human-like paragraphs of text. The concern about the potential misuse of this tremendous generative capability has led to the development of automated systems to detect machine-generated text. These systems can be categorized in: classifiers trained from scratch, classifiers exploiting language distributional features and fine-tuned Neural Language Models [16].

The simplest detectors follow the popular two-step procedure for machine-learning text classifiers, i.e. extracting features from a text excerpt and then feeding them to a machine-learning (ML) or a neural network (NN) classifier. Solaiman et al. (2019) [30] used *tf-idf* features (unigrams and bigrams) and encodings extracted from a GPT-2 model as inputs to a logistic regression model and a simple threshold over a total log probability discriminator, respectively; their aim was to study how different sizes of text generative models and sampling techniques affect the detection. Fagni et al. (2020) [11] additionally explored BERT as the feature generator, followed by Random Forest or SVC as the classifier. Tay et al. (2020) [33] investigated different encoding techniques as well, including the embeddings coming from ConvNet, LSTM and Vaswani et al. (2017)'s Transformer [36]; they outlined that text generators leave artifacts that can be exploited for authorship attribution, as well as to discriminate between human and machine written text. Senait et al. (2021) [34] used Glove [24] and RoBERTa [21] as feature extractors; the word representations were fed to either a three-layer dense NN, a CNN (one embedding layer, one convolutional layer, one global max pooling and a dropout layer), a Long Short-Term Memory (LSTM) network, or a Hierarchical Attention Network (HAN), whose aim is to capture hierarchical structures of text.

Trained from scratch Deep Neural Networks (DNN) may be employed too: Fagni et al. (2020) [11] further tested deep neural networks working at character level, such as *char_cnn, char_gru* and *char_cnngru*, while Uchendu et al. (2020) [35] compared various RNNs and CNNs variants at word level to study the authorship attribution of a generated text; instead, Saravani et al. (2021) [28] trained a network composed by CTBERT-v2 [22], which is a BERT model pre-trained on Covid-19 tweets, followed by a Bidirectional LSTM (BiLSTM) to capture more temporal relations in the sentence, and the NeXtVLAD network [20] to summarize the most important information.

Bakhtin et al. (2019) [4] stepped out from the previous classification paradigms with an energy-based model detector. Zhong et al. (2020) [44] looked at text from a different point of view, developing a graph-based model that utilizes the factual structure of a document. Moreover, Tan et al. (2020) [32]'s system exploited semantic inconsistencies between the text article and the attached images (along with the captions) to defend against machine-generated news articles.

The only works that dealt with language distributional features were carried out by Badaskar et al. (2008) [3] and Gehrmann et al. (2019) [13], respectively. The first studied empirical, syntactic and semantic features over texts sampled from a trigram language model; the second provided GLTR, a visually statistical forensic tool to aid humans in detecting machine-generated text.

Last but not least, detector systems may consist in jointly fine-tuning the original Transformer's architecture (such as GPT-2, GROVER, BERT and RoBERTa) with a final neural network *binary* classifier[1](*human* and *bot* class) over a target dataset (typically smaller than the pre-training one). Usually, satisfying results can be obtained in few epochs. Zellers et al. (2019) [42] fine-tuned GROVER, GPT-2 and BERT over GROVER's generated articles, finding out that GROVER was the best one at detecting GROVER's fake texts. This suggested that the best defense against text generator models may be the generator itself. However, Solaiman et al. (2019) [30] proved it wrong: having GPT-2 texts as the target dataset, fine-tuning a RoBERTa model achieved higher accuracy than fine-tuning a GPT-2 model with equivalent capacity. Fagni et al. (2020) [11] fine-tuned on tweets the most popular Transformer-based language models (BERT, RoBERTa, XLNet, DistilBERT). Stiff et al. (2021) [31] evaluated GROVER and OpenAI RoBERTa based detectors over several datasets comprising a wide variety of machine-generated texts, such as news articles, tweets, forum comments and product reviews; they showed that those publicly available detectors cannot generalize well over texts not seen during training or fine-tuning, and this is especially true on social media posts. Moreover, they demonstrated that GROVER and RoBERTa detectors are not robust to both white and black box adversarial attacks (DeepWordBug[12] was used), whose goal is to make them misclassifying deepfake texts as human written (e.g. by changing some chars).

Adelani et al. (2019) [1] and Tay et al. (2020) [33] evaluated ensemble methods too: the former fused Grover-based detector, GLTR and RoBERTa-based detector from OpenAI using a logistic regression at the score level, while the latter employed authorship attribution techniques using established machine-learning algorithms such as Random Forest.

Finally, to the best of our knowledge, few works concerned the detection of machine-generated text on Social Media: Adelani et al. (2019) [1] dealt with Amazon and Yelp reviews, while Fagni et al. (2020) [11], Senait et al. (2021) [34], Stiff et al. (2021) [31] and Saravani et al. (2021) [28] with tweets. Fagni et al. (2020) 's work [11] is the main study of deepfake text in a *real social media setting*, where the generator is unknown and the text is much shorter than a news article; the authors released TweepFake, the first dataset of machine-generated tweets including those written by the famous GPT-2; together with the dataset, they provided the baseline detectors for this detection task, reaching 90% of accuracy using RoBERTa jointly fine-tuned with a neural network classifier on TweepFake. Compared to this work, our aim was to investigate whether tuning the hyper-parameters during the transfer learning of the Transformer-based detectors adopted by Fagni et al. (2020) [11] brings performance improvement or not (over the GPT-2 tweets in particular); moreover, we tested four other Transformer-based detectors (GPT-2, BART, BERTweet and TwitterRoberta) fine-tuning them with both the default and tuned hyper-parameters over the TweepFake tweets; the stacking ensemble of the most promising three (BART, BERTweet and TwitterRoberta) was investigated as well.

During the last years, TweepFake has been exploited to develop ever more powerful deepfake social media text detection models with respect to the baseline set by Fagni et al. (2020) [11]. First, Senait et al. (2021) 's model [34] reached 87.9% of accuracy by feeding the HAN network with Roberta's word representations and fine-tuning it on a dataset composed by TweepFake and some augmented tweets produced with EDA [38]; on the other hand, our work focused on pushing the limits of a Transformer-based detector, since it is currently the most promising technique over the original TweepFake dataset, reaching 93.6% of accuracy. Second, Stiff et al. (2021) [31] showed that the Open AI sequential classifier based on RoBERTa can generalize (i.e. without fine-tuning on TweepFake) with a 77.6% of accuracy over TweepFake tweets; similarly, we probed our eight Transformer-based detectors (both fine-tuned with default and tuned hyper-parameters over the original TweepFake dataset) over eight GPT-3 Twitter bots to investigate their ability to generalize with respect to a more sophisticated language model evolution (GPT-3) of the generative model on whose generated texts (GPT-2 tweets) the detector has been fine-tuned on. Lastly, Saravani et al. (2021) [28] reached 92% of accuracy by fine-tuning on TweepFake the CTBERT-v2 model followed by a BiLSTM and the text adapted NeXtVLAD. However, the same accuracy was reached also by fine-tuning CTBERT-v2 on TweepFake, indicating that their own complex network's results are only due to the usage of a domain-specific pre-trained language model; taking inspiration from this research, we tested the same BERTweet model [23] as detector, as well as an additional twitter-specific pre-trained language model that is TwitterRoberta [8]: unlike Saravani et al. (2021) [28], we showed the accuracy over the four generative sources of TweepFake dataset: *Human*, *GPT-2*, *RNN* and other older techniques (comprising Markov Chains method) to which Fagni et al. (2020) [11] assigned the same label *Others*.

3 TESTED APPROACHES

This section first describes how the hyper-parameters of the state-of-the-art detectors based on Transformer-based language models [11] are tuned; the results will give hints on their real detection capabilities. Then, the ensemble technique is detailed, followed by the description of the experimental set-up.

As pre-trained language models for our detectors, we chose eight of them in such a way to include auto-regressive (GPT2, [26]; XLNET[2], [41]), bi-directional (BERT, [10]; DistilBERT, [27]; RoBERTa, [21]), encoder-decoder (BART, [19]) and pre-trained on tweets (BERTweet [23]; TwitterRoberta [8]) models. Compared to [28] which used CTBERT-v2 pre-trained on 22M Covid-19 English Tweets, we chose to test two language models pre-trained on a much larger number of tweets, that is 850M English Tweets for BERTweet and 58M for TwitterRoberta; BERTweet has got the same BERT's

[1] It is nothing more than a binary sequence classifier

[2] It can be considered bidirectional as well, being a Permutational Language Model.

architecture, but it's pre-trained using the optimization procedure of RoBERTa; on the other hand, TwitterRoberta is a plain RoBERTa base model pre-trained on tweets. See Table 1 for the pre-trained models' details. Remember that our detectors are Transformer-based binary sequence classifiers, i.e. labelling a tweet as *human* or *bot* written.

Table 1: The chosen pre-trained Transformers.

Transformer	Details of the model			
	Layers	Emb. size	Att. heads	Tot param.
gpt2_small	12	768	12	117M
bart_facebook_large	24	1024	16	406M
bert_base_uncased	12	768	12	110M
distilbert_base_uncased	6	768	12	66M
roberta_base	12	768	12	125M
xlnet_base_cased	12	768	12	110M
bertweet	12	768	12	135M
twitter_roberta_base	12	768	12	125M

3.1 Hyper-parameters Tuning

Typically, several hyper-parameters can be calibrated for fine-tuning a Transformer-based sequence classifier over a target dataset: the number of training epochs, the mini-batch's size, the learning rate for the optimization algorithm (AdamW works generally fine), the weight decay for regularization purposes, and the warmup ratio for the Slanted Triangular Learning Rates (STRL) scheduler (commonly used in Transformers). Due to GPU's memory limits, the mini-batch's size was fixed to 8. Also, the hyper-parameter tuning process had to be divided into two phases due to time constraints: first we tuned the number of training epochs fixing the other hyper-parameters to their default value[3], then we calibrated the remaining hyper-parameters having the number of training epochs fixed instead. For the first phase, a grid search method was enough; the second phase, still due to time limits, employed the bayesian optimization with the parameters shown in Table 2. In both tuning phases, we selected the hyper-parameters setting that brought to the highest *evaluation* accuracy. All tuning phases' info are shown in Table 2. We applied the two phases to all our Transformer-based detectors. We also had to evaluate the GPT-2, BART, BERTweet and TwitterRoberta based detectors with the default values[4], as [11] didn't evaluate these methods. Notice that the default values for the hyper-parameters to tune fell into the range of the explored values during the two tuning phases.

Even though we want to test the detectors' capabilities over human and GPT-2 tweets, we didn't want to bias the results: for this reason, we tuned the hyper-parameters over the entire TweepFake dataset, which comprises tweets written by older generative techniques as well (see Section 4). This was also a chance to improve the state-of-the-art of deepfake tweet detection task.

3.2 Ensemble Learning

The single-level *stacking* learner [40] was chosen as the ensemble method, as we didn't have to neither choose voting weights, nor generate new samples. *Stacking* learns heterogeneous base learners in parallel, and combines them by training a meta-model to output a prediction based on the different base models predictions. The training and validation sets (see Section 4) were combined in a larger training set, which was used to train *all* the base learners in a *10-fold cross-training* manner. This learning step produced a new training set, where each training tweet has got the corresponding prediction for each base learner. The obtained training set was divided into new training and validation sets to tune the hyper-parameters of the meta-learner with the grid-search method (see Table 3). The trained meta-learner was tested over the hold-out test set. As base learners, we chose the best three tuned Transformer-based detectors according to their *evaluation* global accuracy. *SVC* was picked as the meta-learner, being the best machine-learning classifier over tweets [11].

3.3 Evaluation Metrics

The used evaluation measures are the ones typically adopted in text classification context: the precision, recall and F1 for each class label. Given that the TweepFake dataset is balanced with respect to the *human* and *bot* classes (see Section 4), we reported also the accuracy. In particular, we evaluated the accuracy of our detectors on the five account categories: *Human*, *GPT-2*, *RNN*, *Others* and *GPT3* (see Section 4 for further details). Each accuracy reveals how much the detector is accurate in identifying the *Human/GPT2/RNN/Others/GPT3* tweets as machine-generated (*bot*). These specific accuracies were computed as described by [11].

3.4 Software and Hardware

The `SimpleTransformers` Python library[5] was used to implement our Transformer-based detectors, which are nothing more than the original Transformers' architecture followed by a neural network binary classifier. The GPT-2 detector was trained on both the classification and the language modeling objectives as described by [25]; the language modeling coefficient λ was set to 0.5 for all GPT2's experiments.

All experiments were conducted on Google Colab (public version), which provided us with Tesla T4 and Tesla P100-PCIe GPUs.

4 DATASET

We conducted our experiments on the original [11]'s *TweepFake* dataset[6]. It consists of tweets coming from 23 bot and 17 human accounts. Each account has got a coarse-grained (*human* or *bot*) and a fine-grained label. The latter indicates the employed text generation technique, namely *human* (17 accounts, 12786 tweets), *GPT-2* (11 accounts, 3861 tweets), *RNN* (7 accounts, 4181 tweets) or *Others* (5 accounts, 4876 tweets). *Others* refers to methods either non-better specified or found in a very low number of accounts (Markov Chain, RNN + Markov Chain, LSTM, CharRNN). The 25572 tweets are split into train validation and validation and test sets. All sets are balanced with respect to the *human* and *bot* tweets.

[3] All hyper-parameters' default value can be found in the `SimpleTransformers` library's documentation at https://SimpleTransformers.ai/docs/usage/

[4] Following [11]'s experimental setup, we limited the number of training epochs to three.

[5] https://github.com/ThilinaRajapakse/SimpleTransformers

[6] https://www.kaggle.com/mtesconi/twitter-deep-fake-text

Table 2: Settings for the two tuning phases. All the unspecified hyper-parameters are left to their default values, as defined in the `SimpleTransformers` Python library.

Phase	Tuning technique				Tuning params		
	Method	Search Params			Param		Values
1	grid search	-			#_training_epochs		[1, 10]

2	Bayesian Optimization	metric	name
			goal
				accuracy
				maximize
					AdamW	learning_rate
						weight_decay
							{"min": 0.0, "max": 1.5e-4}
							{"min": 0.0, "max": 0.1}

		early terminate	type min_iter	hyperband 6	STRL[a]	warmup_ratio	{"min": 0.0, "max": 0.1}
		runs	max	30			

[a] Slanted Triangular Learning Rate

Table 3: SVC meta-learner tuning hyper-parameters.

Parameter	Values
kernel	rbf,linear
rbf_gamma	$[1e-3, 1e-4]$
c	[1, 10, 100, 1000]

Instead, the probing of the detectors' capabilities over GPT-3 social media texts was carried out over 3795 original tweets (no retweets) written by eight Twitter bots based on GPT-3. We preprocessed tweets in such a way to extract just the generated $< text >$, since some accounts framed the GPT-3 texts in a template like " $< text >$ "$\#GPT3$ or $< inventedword >$:$< text >$. Table 4 summarizes the number of tweets for each collected GPT-3 account.

Table 4: Number of tweets of each GPT-3 Twitter account.

bot_1	bot_2	bot_3	bot_4	bot_5	bot_6	bot_7	bot_8
1944	1061	390	222	101	41	28	7

5 EXPERIMENTAL RESULTS AND DISCUSSION

First we show the results and discuss about the improvement of the deepfake tweet detection task over GPT-2 tweets using hyper-parameter tuning and the stacking ensemble technique; we also debate on the detection in a real-setting scenario. Lastly, we discuss the findings on the detectors' capabilities over GPT-3 tweets and the overall improvement of the deepfake tweet detection task. Notice that we re-implemented [11]'s Transformer-based detectors and fine-tuned them over the TweepFake dataset, as we wanted to test those detectors on GPT-3 tweets as well.

5.1 Improvement of GPT-2 tweets detection

Table 5 recaps the default and found tuned hyper-parameters, while Table 6 show the global performance of default and hyper-parameter-tuned versions of the tested Transformer-based detectors.

Table 5: The default and found tuned hyper-parameters for each Transformer-based detector and the stacking ensemble. Approximated values are shown.

Method	Tr. Epochs	Learning Rate	Weight Decay	Warmup Ratio
<method>_default_ft	3	4e-5	0.0	6e-2
bert_opt_ft	3	3.19e-5	9.44e-2	7.86e-2
distilbert_opt_ft	7	3.18e-5	6.61e-2	3.43e-2
roberta_opt_ft	7	4.95e-6	6.29e-2	2.34e-2
xlnet_opt_ft	9	2.42e-5	4.95e-2	1.77e-2
gpt2_opt_ft	3	4.87e-5	2.75e-2	1.49e-2
bart_opt_ft	1	1.39e-5	5.94e-2	6.15e-2
bertweet_opt_ft	5	1.59e-5	9.95e-2	9.67e-2
twitter_roberta_opt_ft	5	1.76e-5	2.56e-2	8.87e-4
Method	Kernel		rbf_gamma	C
ensemble	rbf		1e-4	1

Generally, the goal of hyper-parameter tuning in a classification task is to balance the performance over each class (here *human* and *bot*); this is confirmed by the accuracy results in Table 6: apart from BERT whose optimal hyper-parameters remained the default ones, every other detector balanced itself by either increasing or decreasing the accuracy on the detection of human tweets, while doing the opposite on machine-written ones (GPT-2 + RNN + Others tweets). However, the increase on the human side was more frequent. Notice that also among the bot categories the balance phenomenon can be highlighted. As [28] pointed out, using a Transformer-based language model pre-trained on tweets as the base for a deepfake tweet detector is currently the best option; probably it's due to its increased ability in capturing the hidden features of human-written tweets, as shown in Table 6 for both the default and optimized versions of BERTweet and TwitterRoberta based detectors. In particular, default and optimized BERTweet-based detectors performed better than TwitterRoberta-based ones on both human and GPT-2 tweets, meaning that pre-training on a much more large amount of tweets is worth it; moreover, BERTweet-based detector is the only one whose hyper-parameter tuning boosted the GPT-2 accuracy by almost 6%: the latent features of GPT-2 tweets discovered by the optimized BERTweet-based detector are the most significant.

Table 6: Comparison of the accuracy over different types of account. (+), (−) or (=) indicates the accuracy variation with respect to the detector with default hyper-parameters, while the bold accuracies highlight the best detectors.

Method default_ft	Global	Human	GPT-2	RNN	Others
bert[a]	0.891	0.871	0.737	0.998	0.948
distilbert[a]	0.887	0.883	0.732	0.993	0.942
roberta[a]	0.896	0.893	0.740	0.995	0.952
xlnet[a]	0.877	0.855	0.781	0.990	0.975
			OURS		
gpt2	0.902	0.883	0.794	0.995	0.959
bart	0.901	0.904	0.714	0.998	0.965
bertweet	0.916	0.928	0.747	0.995	0.946
twitter_ roberta	0.915	0.906	0.789	0.998	0.971
Method opt_ft					
bert	0.891(=)	0.871(=)	0.737(=)	0.998(=)	0.948(=)
distilbert	0.885(−)	0.879(−)	0.721(−)	**1.000**(+)	0.948(+)
roberta	0.907(+)	0.900(+)	0.747(+)	0.993(−)	0.963(+)
xlnet	0.889(+)	0.887(+)	0.755(−)	0.995(+)	0.946(−)
gpt2	0.905(+)	0.889(+)	0.799(+)	0.995(=)	0.961(+)
bart	0.904(+)	0.906(+)	0.724(+)	0.993(−)	**0.971**(+)
bertweet	**0.936**(+)	**0.947**(+)	0.802(+)	0.995(=)	0.967(+)
twitter_ roberta	0.920(+)	0.924(+)	0.784(−)	0.998(=)	0.948(−)
ensemble	0.934	0.932	**0.844**	0.998	0.961

[a] our re-implementation of BERT, DistilBERT, RoBERTa and XLNET based detectors with default hyper-parameters as described by Fagni et al. (2020) [11]

As far as it concerns the ensemble, we identified the optimized BERTweet, TwitterRoberta and BART based detectors as the base learners, as they were the top-three methods over the validation set; the optimized BERTweet and TwitterRoberta based detectors where obviously included in the ensemble, being the best ones at recognizing human tweets, in addition to BERTweet being the preferred choice to recognize GPT-2 tweets as machine-generated. Besides, BART-based detector may be significant for its particular encoder-decoder architecture, differently from the other Transformer-based language models which accounts for only encoders or decoders. Ultimately, the ensemble found a compromise by lowering the accuracy on human tweets, while greatly increasing it on GPT-2 ones and reaching the best accuracy of 84.4%.

5.2 Analysis in a real-setting scenario

In a real setting scenario, what matters is having a low alarming rate (e.g.,< 10%) but a high true positive rate; in other words, a detector should incorrectly label a human tweet as bot *as little as* possible, but correctly recognizing deepfake tweets most of the times. To this aim, we considered the *bot* class as the *positive* class, and we computed our detectors' ROC curves. The *FPR* is the detector's alarm rate. The higher the detector's *TPR* at low alarming rates, the better. Notice that the implicit decision threshold for the reported detection accuracies is 0.5 over the probability of the *bot* class. Moreover, since we want to probe the detectors' capabilities in recognizing GPT-2 tweets as fake, we plotted the ROC curves considering just the human and GPT-2 tweets (thus excluding the RNN and Others tweets of TweepFake dataset). The ROC curves (Figure 1) shows that the ensemble detector is the best choice for GPT-2 tweets detection in a real-setting scenario, where the alarming rate is low: the single Transformer-based detectors had troubles in reaching the ensemble's true positive rates ($TPR > 80\%$), particularly for alarming rates starting from 7%. For very low alarming rates ($FPR <$ 7%), BERTweet detector seemed the best picking; however, in that zone the TPR decreases dramatically, and since the number of human tweets in the test set is 1278 (with 1280 bot tweets), a FPR of 1% or 0.1% evaluates the corresponding TPR on just 12 or one tweets. Thus, values of TPR for very small FPR are less statistically significant.

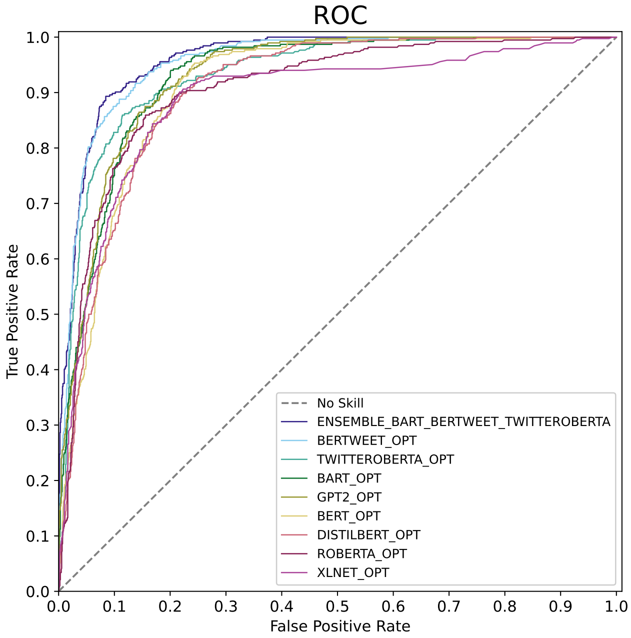

Figure 1: ROC curves for the optimized Transformer-based detectors and the ensemble, computed over Human and GPT-2 tweets only.

5.3 GPT-3 tweets detection

Table 7 shows the accuracy of each tested detector over GPT-3 tweets, as well as the specific one for each GPT-3 bot; remind that the detectors were fine-tuned over the TweepFake dataset which contains GPT-2 tweets but not GPT-3 ones, hence those GPT-3 accuracy values indicate the detectors' ability to generalize with respect to a more sophisticated and complex language model evolution of the generative model on whose generated texts the detector has been fine-tuned on.

Generally, each detector did not generalize well over GPT-3 tweets, since their accuracy significantly decreased over them with respect to the results over TweepFake tweets. Remarkably, those detectors that recognized human-written tweets very skillfully (BERTweet, TwitterRoberta and the ensemble) decreased their GPT-3 accuracy to random guess; this demonstrates that social media texts generated by GPT-3 are more similar to human-written ones than those produced by GPT-2, highlighting that detectors fine-tuned on an advanced generative technique like GPT-2 cannot keep up the pace with a more complex one like GPT-3.

Moreover, XLNET-based detector with default hyper-parameters was surprisingly the best at detecting GPT-3 tweets over all collected GPT-3 bots (82.1% of accuracy), as Table 7 shows; the only exception is bot_8, but it is not significant since it has got only seven tweets. Presumably, it indicates that XLNET, unlike the other detection methods, learns latent features which better describe machine-written tweets instead of human ones; besides, it produces the highest difference between human and bot accuracies. Linked to this finding, we can state that if on the one hand the GPT-3 social media texts are much closer to texts produced by humans, on the other hand they must have (hidden) peculiarities in common share with other generative techniques (like GPT-2, RNN and Other older methods) that the other tested detectors have considered with lower priority. Notice that the optimized XLNET-based detector decreased its accuracy on GPT-3 tweets (71.8% of accuracy): since the optimization phase tries to increase the balance between the human and bot classes, XLNET found this balance in increasing the accuracy over human tweets while decreasing it over bot tweets, giving up precious bot features.

Observing both the most accurately detected bot's tweets (bot_7) and the less accurately detected bot's ones (bot_8), we noticed that bot_8 contains Twitter user mentions (mostly related to reply tweets) and it writes longer tweets than bot_7, which posts brief original tweets (without mentions). Inspecting all GPT3 bots' tweets and human written tweets, we observed that our detectors classified as bot those tweets having a low number of words (around 12 words for human tweets and 16 words for GPT3 ones), no mentions or a bunch of them in the middle of the text (no reply tweets), and no urls. On the other hand, our detectors labelled as human those tweets having a high number of words (around 18 words for human tweets and 23 words for GPT3 ones); human tweets correctly labelled as human contained from zero to one url and/or around two mentions, the latter usually located at the beginning of the text.

Table 8 shows misclassified examples of Human, GPT-2 and GPT-3 tweets.

Table 7: Comparison of the accuracy over the eight GPT-3 Twitter accounts.

Method default_ft	all *bots*	bot_1	bot_2	bot_3	bot_4	bot_5	bot_6	bot_7	bot_8
bert[a]	0.614	0.652	0.614	0.669	0.680	0.545	0.548	0.821	0.143
distilbert[a]	0.613	0.648	0.577	0.615	0.509	0.515	0.643	0.786	0.286
roberta[a]	0.608	0.598	0.588	0.718	0.622	0.515	0.690	0.750	0.143
xlnet[a]	**0.821**	0.818	0.826	0.928	0.694	0.723	0.801	0.821	0.143
gpt2	0.736	0.736	0.776	0.841	0.482	0.554	0.643	0.821	0.143
bart	0.520	0.559	0.396	0.649	0.505	0.574	0.571	0.786	0.000
bertweet	0.690	0.717	0.745	0.649	0.455	0.366	0.476	0.786	0.429
twitter roberta	0.549	0.582	0.449	0.667	0.482	0.614	0.548	0.714	0.429
Method opt_ft									
bert	0.614	0.693	0.820	0.718	0.595	0.535	0.548	0.929	0.000
distilbert	0.618	0.644	0.548	0.692	0.608	0.624	0.619	0.750	0.000
roberta	0.621	0.629	0.633	0.605	0.514	0.624	0.619	0.786	0.286
xlnet	0.718	0.724	0.714	0.831	0.559	0.624	0.643	0.714	0.000
gpt2	0.734	0.722	0.815	0.772	0.500	0.515	0.667	0.893	0.000
bart	0.531	0.617	0.318	0.685	0.541	0.475	0.619	0.714	0.000
bertweet	0.557	0.635	0.484	0.497	0.401	0.406	0.500	0.714	0.143
twitter roberta	0.525	0.579	0.375	0.631	0.554	0.535	0.595	0.643	0.571
ensemble	0.545	0.627	0.376	0.636	0.495	0.465	0.595	0.714	0.143

[a] our re-implementation of BERT, DistilBERT, RoBERTa and XLNET based detectors with default hyper-parameters as described by Fagni et al. (2020) [11].

Table 8: Examples of Human, GPT-2 and GPT-3 tweets.

Tweet text	Written by	Labelled as
twitter hiring sherlock holmes to prowl the office wih a magnifying glass and find the employee who posted "Aids piss" on the pizza hut acct	Human	Bot
What is going on in Iowa besides all of the money wasted. They don't want to host the WH, who is having its best economic year in many years.	GPT-2	Human
If you're working hard and not achieving the results you want, the only thing you can change is yourself. So change yourself.	GPT-3	Human

5.4 Improvement of the state-of-the-art deepfake tweets detection task

Table 9 shows the general performance of the tested detectors against the state-of-the-art ones [11, 28, 31, 34] over the TweepFake dataset.

The BERT-based detector was the only one whose hyper-parameters' default value was already the optimized one according to our bayesian optimization on the validation set. Every other detector generally improved its global and specific performances over the test set, except for the one based on DistilBERT. This exception may be due to the fact that, unlike the other tested Transformer-based language models, DistilBERT's aim was not to improve a language model understanding capabilities, but to reduce the model's size and computational time. Hence DistilBERT is not optimized to further boost language modeling.

Table 9: Comparison of our optimized Transformer-based detectors and ensemble with the best state-of-the-art results from [11, 28, 31, 34]. (+) or (−) indicates whether the optimized version increased the evaluation metric or not. Results in bold indicate the best values among all detectors.

Method		human			bot		globally
	precision	recall	F1	precision	recall	F1	accuracy
bert_default_ft[a]	0.899	0.882	0.890	0.884	0.901	0.892	0.891
distilbert_default_ft[a]	0.894	0.880	0.886	0.882	0.895	0.888	0.887
roberta_default_ft[a]	0.901	0.890	0.895	0.891	0.902	0.897	0.896
xlnet_default_ft[a]	0.914	0.832	0.871	0.846	0.922	0.882	0.877
roberta + HAND[b]	n/a	n/a	n/a	n/a	n/a	n/a	0.897
CTBERT-v2+BiLSTM+NeXtVLAD (Domain-FT) Cfg 1[c]	0.92-	0.91-	0.92-	0.92-	0.92-	0.92-	0.92-
CTBERT-v2 (Domain-FT) Cfg 3[c]	0.91-	0.92-	0.92-	0.92-	0.91-	0.92-	0.92-
OpenAI roberta large[d]	n/a	n/a	n/a	n/a	n/a	n/a	0.776
OURS							
gpt2_default_ft	0.918	0.883	0.900	0.887	0.921	0.904	0.902
bart_default_ft	0.900	0.902	0.901	0.902	0.900	0.901	0.901
bertweet_default_ft	0.905	0.93	0.917	0.928	0.902	0.915	0.916
twitter_roberta_default_ft	0.923	0.905	0.914	0.907	0.925	0.915	0.915
bert_opt_ft	0.899(=)	0.882 (=)	0.890 (=)	0.884 (=)	0.901 (=)	0.892 (=)	0.891 (=)
distilbert_opt_ft	0.894 (=)	0.873 (−)	0.884 (−)	0.876 (−)	0.897 (+)	0.887 (−)	0.885 (−)
roberta_opt_ft	0.908 (+)	0.906 (+)	0.907 (+)	0.906 (+)	0.908 (+)	0.907 (+)	0.907 (+)
xlnet_opt_ft	0.902 (−)	0.874 (+)	0.888 (+)	0.878 (+)	0.905 (−)	0.891 (+)	0.889 (+)
gpt2_opt_ft	0.920 (+)	0.889 (+)	0.903 (+)	0.890 (+)	0.923 (+)	0.906 (+)	0.905 (+)
bart_opt_ft	0.904 (+)	0.904 (+)	0.904 (+)	0.904 (+)	0.904 (+)	0.904 (+)	0.904 (+)
bertweet_opt_ft	0.928 (+)	**0.945** (+)	**0.936** (+)	**0.944** (+)	0.927 (+)	**0.935** (+)	**0.936** (+)
twitter_roberta_opt_ft	0.916 (−)	0.924 (+)	0.92 (+)	0.924 (+)	0.915 (−)	0.919 (+)	0.920 (+)
ensemble	**0.937**	0.930	0.934	0.931	**0.936**	0.934	0.934

[a] our re-implementation of BERT, DistilBERT, RoBERTa and XLNET based detectors with default hyper-parameters as described by Fagni et al. (2020) [11].
[b] fine-tuned on TweepFake + Augmented tweets as described by Senait et al. (2021) [34]
[c] as described in Saravani et al. (2021) [28]
[d] without fine-tuning on TweepFake tweets, as described by Stiff et al. (2021) [31]

Our two best detectors, i.e. BERTweet and the Ensemble, surpassed the state-of-the-art fine-tuned detector based on a BERT model pre-trained on 22M Covid-19 English Tweets [28]; this demonstrates that it is better to employ language models pre-trained on general English Tweets. This is true when dealing with GPT-2 tweets or older generative techniques, but the situation overturns when GPT-3 social media texts come into play: new detection approaches must be devised to take into account GPT-3 texts as well.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we studied and improved the performance of the state-of-the-art deepfake tweet detection methods over GPT-2 tweets, where each detector was fine-tuned on a dataset comprising GPT-2 examples; afterwards, we analyzed the detectors' ability to generalize on tweets written by the more sophisticated and complex evolution of GPT-2, that is GPT-3. Results showed that the class balance brought by the hyper-parameter optimization of a Transformer-based language model followed by a neural network classifier may either increase or decrease the accuracy on detecting GPT-2 tweets as machine-generated; undoubtedly, BERTweet, a language model pre-trained on a large amount of English Tweets, is the optimal base for a detector, reaching 94.7% of accuracy on human-written tweets and 80.2% on GPT-2 ones. Moreover, the SVC stacking ensemble comprising BERTweet, BART and TwitterRoberta based detectors increased the GPT-2 accuracy to 84.4%; it was also the best detection method in a real-setting scenario, since it outperformed every other tested transformer-based detectors (with $TPR > 80\%$) starting from a low alarming rate (FPR) of 7%. All in all, the hyper-parameter optimization and stacking ensemble advanced the state-of-the-art on the deepfake tweet detection task. Nonetheless, all tested detectors did not generalize well on GPT-3 tweets; noticeably, even though BERTweet-based detector and the ensemble were the best at correctly recognizing human tweets as human-written and GPT-2 ones as machine-generated, they decreased their accuracy on GPT-3 tweets to random guess. This suggests that GPT-3 tweets are very similar to human ones. However, also GPT-3 social media texts have got latent features that discriminate them from human posts, as XLNET proved with an accuracy of 82.1%. This findings demonstrate that malicious users can potentially already contribute to the information disorder on social media with GPT-3 short texts without being detected by the state-of-the art deepfake social media text detectors. With this in mind, we call for further assessments over the detection of GPT-3 social media texts (not restricted to Twitter only).

REFERENCES

[1] David Ifeoluwa Adelani, Haotian Mai, Fuming Fang, Huy H. Nguyen, Junichi Yamagishi, and Isao Echizen. 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, Francesco Moscato, Tomoya Enokido, and Makoto Takizawa (Eds.). Springer International Publishing, Cham, 1341–1354.

[2] Murat Ayfer. 2020. *Philosopher AI.* Retrieved February 15, 2022 from https://philosopherai.com/

[3] Sameer Badaskar, Sachin Agarwal, and Shilpa Arora. 2008. Identifying Real or Fake Articles: Towards better Language Modeling. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II.*

[4] Anton Bakhtin, Sam Gross, Myle Ott, Yuntian Deng, Marc'Aurelio Ranzato, and Arthur Szlam. 2019. Real or fake? learning to discriminate machine from human generated text. arXiv:1906.03351 [Preprint].

[5] Alessandro Bessi and Emilio Ferrara. 2016. Social bots distort the 2016 US Presidential election online discussion. *First Monday* 21, 11 (2016).

[6] Samantha Bradshaw, Hannah Bailey, and P Howard. 2021. Industrialized disinformation: 2020 global inventory of organized social media manipulation. Computational Propaganda Research Project.

[7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 1877–1901. https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[8] Hugginface cardiffnlp. 2021. *Twitter-roBERTa-base.* Retrieved February 10, 2022 from https://huggingface.co/cardiffnlp

[9] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2017. The Paradigm-Shift of Social Spambots: Evidence, Theories, and Tools for the Arms Race. In *Proceedings of the 26th International Conference on World Wide Web Companion* (Perth, Australia) *(WWW '17 Companion)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 963–972. https://doi.org/10.1145/3041021.3055135

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[11] Tiziano Fagni, Fabrizio Falchi, Margherita Gambini, Antonio Martella, and Maurizio Tesconi. 2021. TweepFake: About detecting deepfake tweets. *PLOS ONE* 16 (05 2021), 1–16. https://doi.org/10.1371/journal.pone.0251415

[12] Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-Box Generation of Adversarial Text Sequences to Evade Deep Learning Classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*. 50–56. https://doi.org/10.1109/SPW.2018.00016

[13] Sebastian Gehrmann, Hendrik Strobelt, and Alexander Rush. 2019. GLTR: Statistical Detection and Visualization of Generated Text. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Association for Computational Linguistics, Florence, Italy, 111–116. https://doi.org/10.18653/v1/P19-3019

[14] Karen Hao. 2020. *A college kid's fake, ai-generated blog fooled tens of thousands. this is how he made it.* Retrieved February 10, 2022 from https://www.technologyreview.com/2020/08/14/1006780/ai-gpt-3-fake-blog-reached-top-of-hacker-news/

[15] Will Douglas Heaven. 2020. *A GPT-3 bot posted comments on Reddit for a week and no one noticed.* Retrieved February 14, 2022 from https://www.technologyreview.com/2020/10/08/1009845/a-gpt-3-bot-posted-comments-on-reddit-for-a-week-and-no-one-noticed/

[16] Ganesh Jawahar, Muhammad Abdul-Mageed, and Laks Lakshmanan, V.S. 2020. Automatic Detection of Machine Generated Text: A Critical Survey. In *Proceedings of the 28th International Conference on Computational Linguistics*. International Committee on Computational Linguistics, Barcelona, Spain (Online), 2296–2309. https://doi.org/10.18653/v1/2020.coling-main.208

[17] Jeff Kao. 2017. *More than a Million Pro-Repeal Net Neutrality Comments were Likely Faked | Hacker Noon.* Retrieved February 10, 2022 from https://medium.com/hackernoon/more-than-a-million-pro-repeal-net-neutrality-comments-were-likely-faked-e9f0e3ed36a6

[18] Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. arXiv:1909.05858 [Preprint].

[19] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 7871–7880. https://doi.org/10.18653/v1/2020.acl-main.703

[20] Rongcheng Lin, Jing Xiao, and Jianping Fan. 2018. NeXtVLAD: An Efficient Neural Network to Aggregate Frame-Level Features for Large-Scale Video Classification. In *Computer Vision - ECCV 2018 Workshops - Munich, Germany, September 8-14, 2018, Proceedings, Part IV (Lecture Notes in Computer Science, Vol. 11132)*, Laura Leal-Taixé and Stefan Roth (Eds.). Springer, 206–218. https://doi.org/10.1007/978-3-030-11018-5_19

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

[22] Martin Müller, Marcel Salathé, and Per Egil Kummervold. 2020. COVID-Twitter-BERT: A Natural Language Processing Model to Analyse COVID-19 Content on Twitter. *ArXiv* abs/2005.07503 (2020).

[23] Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English Tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. 9–14.

[24] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543. https://www.aclweb.org/anthology/D14-1162/

[25] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. (2018). https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf

[26] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. In: OpenAI Blog [Internet]. , 9 pages. https://openai.com/blog/better-language-models/

[27] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv:1910.01108 [Preprint].

[28] Sina Mahdipour Saravani, Indrajit Ray, and Indrakshi Ray. 2021. Automated Identification of Social Media Bots Using Deepfake Text Detection. In *Information Systems Security*, Somanath Tripathy, Rudrapatna K. Shyamasundar, and Rajiv Ranjan (Eds.). Springer International Publishing, Cham, 111–123.

[29] Chengcheng Shao, Giovanni Luca Ciampaglia, Onur Varol, Kai-Cheng Yang, Alessandro Flammini, and Filippo Menczer. 2018. The spread of low-credibility content by social bots. *Nature communications* 9, 1 (2018), 1–9.

[30] Irene Solaiman, Miles Brundage, Jack Clark, Amanda Askell, Ariel Herbert-Voss, Jeff Wu, Alec Radford, and Jasmine Wang. 2019. Release strategies and the social impacts of language models. arXiv:1908.09203 [Preprint].

[31] Harald Stiff and Fredrik Johansson. 2021. Detecting computer-generated disinformation. *International Journal of Data Science and Analytics* (2021), 1–21.

[32] Reuben Tan, Bryan Plummer, and Kate Saenko. 2020. Detecting Cross-Modal Inconsistency to Defend Against Neural Fake News. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Online, 2081–2106. https://doi.org/10.18653/v1/2020.emnlp-main.163

[33] Yi Tay, Dara Bahri, Che Zheng, Clifford Brunk, Donald Metzler, and Andrew Tomkins. 2020. Reverse Engineering Configurations of Neural Text Generation Models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 275–279. https://doi.org/10.18653/v1/2020.acl-main.25

[34] Senait G. Tesfagergish, Robertas Damaševičius, and Jurgita Kapočiūtė-Dzikienė. 2021. Deep Fake Recognition in Tweets Using Text Augmentation, Word Embeddings and Deep Learning. In *Computational Science and Its Applications – ICCSA 2021*, Osvaldo Gervasi, Beniamino Murgante, Sanjay Misra, Chiara Garau, Ivan Blečić, David Taniar, Bernady O. Apduhan, Ana Maria A. C. Rocha, Eufemia Tarantino, and Carmelo Maria Torre (Eds.). Springer International Publishing, Cham, 523–538.

[35] Adaku Uchendu, Thai Le, Kai Shu, and Dongwon Lee. 2020. Authorship Attribution for Neural Text Generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Online, 8384–8395. https://doi.org/10.18653/v1/2020.emnlp-main.673

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 6000–6010.

[37] Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science* 359, 6380 (2018), 1146–1151.

[38] Jason Wei and Kai Zou. 2019. EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 6382–6388. https://doi.org/10.18653/v1/D19-1670

[39] Max Weiss. 2019. *Deepfake Bot Submissions to Federal Public Comment Websites Cannot Be Distinguished from Human Submissions*. Retrieved February 10, 2022 from https://techscience.org/a/2019121801/

[40] David H. Wolpert. 1992. Stacked generalization. *Neural Networks* 5, 2 (1992), 241–259. https://doi.org/10.1016/S0893-6080(05)80023-1

[41] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Curran Associates Inc., Red Hook, NY, USA, Article 517, 11 pages.

[42] Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against Neural Fake News. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Curran Associates Inc., Red Hook, NY, USA, Article 812, 12 pages.

[43] Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. *Grover - A State-of-the-Art Defense against Neural Fake News*. https://grover.allenai.org/ (Accessed on 05/27/2020).

[44] Wanjun Zhong, Duyu Tang, Zenan Xu, Ruize Wang, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. 2020. Neural Deepfake Detection with Factual Structure of Text. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Online, 2461–2470. https://doi.org/10.18653/v1/2020.emnlp-main.193