Know Thine Enemy: Adaptive Attacks on Misinformation Detection Using Reinforcement Learning

Piotr Przybyła[1,2] and **Euan McGill**[1] and **Horacio Saggion**[1]
[1] LaSTUS Lab, TALN Group, Universitat Pompeu Fabra, Barcelona, Spain
[2] Institute of Computer Science, Polish Academy of Sciences, Warsaw, Poland
{piotr.przybyla, euan.mcgill, horacio.saggion}@upf.edu

Abstract

We present XARELLO: a generator of adversarial examples for testing the robustness of text classifiers based on reinforcement learning. Our solution is adaptive, it learns from previous successes and failures in order to better adjust to the vulnerabilities of the attacked model. This reflects the behaviour of a persistent and experienced attacker, which are common in the misinformation-spreading environment. We evaluate our approach using several victim classifiers and credibility-assessment tasks, showing it generates better-quality examples with less queries, and is especially effective against the modern LLMs. We also perform a qualitative analysis to understand the language patterns in the misinformation text that play a role in the attacks.

1 Introduction

Nowadays, an ever-increasing proportion of the text we read online is published by anonymous or unfamiliar authors, e.g. in online news outlets, blogs, social media portals, instant messaging, and communication agents. This puts a great burden on the entities hosting such platforms, having to filter the user-generated data to remove or de-prioritise content considered inflammatory, misleading, unpleasant or simply illegal. A large part of this work is performed manually by moderators, but the use of automatic machine-learning (ML) classifiers is becoming more common (Singhal et al., 2022). This scenario necessitates testing the *robustness* of the deployed models, i.e. their ability to deliver correct results even when their input is manipulated, e.g. by a fake news spreader.

The robustness is usually tested by analysing input examples and checking what kind of modifications made to them confuse the victim classifier to change its output. For example, let us assume the following statement is correctly identified by a classifier as misleading: *Drinking orange juice causes DEATH!*. However, if the same classifiers return a different result when *causes* is replaced with *provokes* or *cuases*, this weakness can be used by attackers. Discovering such *adversarial examples* (AE) is the best way to understand the vulnerabilities of the common methods before they can be exploited by attackers. A plethora of approaches for AE generation for text classifiers has been proposed (Zhang et al., 2020) and tested, including in misinformation detection (Przybyła et al., 2023).

The AE techniques explored so far are usually based on making incremental changes to an individual example (e.g. word replacements), and testing the victim's response to the modifications, until it returns a desired response (Zhang et al., 2020). This simple procedure is repeated for each example independently. Here we consider a different approach, where an attacker is *adaptive* and it learns from successes and failures from each attack attempt. Thus, the attacker can observe and exploit the weaknesses of the victim, i.e. modifications that are particularly likely to flip the classification decision. This corresponds to the real-world circumstances of misinformation spreaders that are established large-scale enterprises, e.g. Russia's Internet Research Agency (DiResta et al., 2019), able to gather significant expertise regarding the weaknesses of the moderation on major platforms.

To understand the effectiveness of such attacks, we propose XARELLO (eXploring Adversarial examples using REinforcement Learning Optimisation), a method for learning weaknesses of a target classifier to improve quality of the proposed modifications. XARELLO is built upon the reinforcement learning framework, which allows it to gather experience in the *adaptation* phase and then use it in the *attack* phase. Using the framework for testing AE solutions in several misinformation detection tasks for English (Przybyła et al., 2023), we show that our solution indeed manages to adapt over time and deliver performance beating the state of the

Proceedings of the 14th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis, pages 125–140

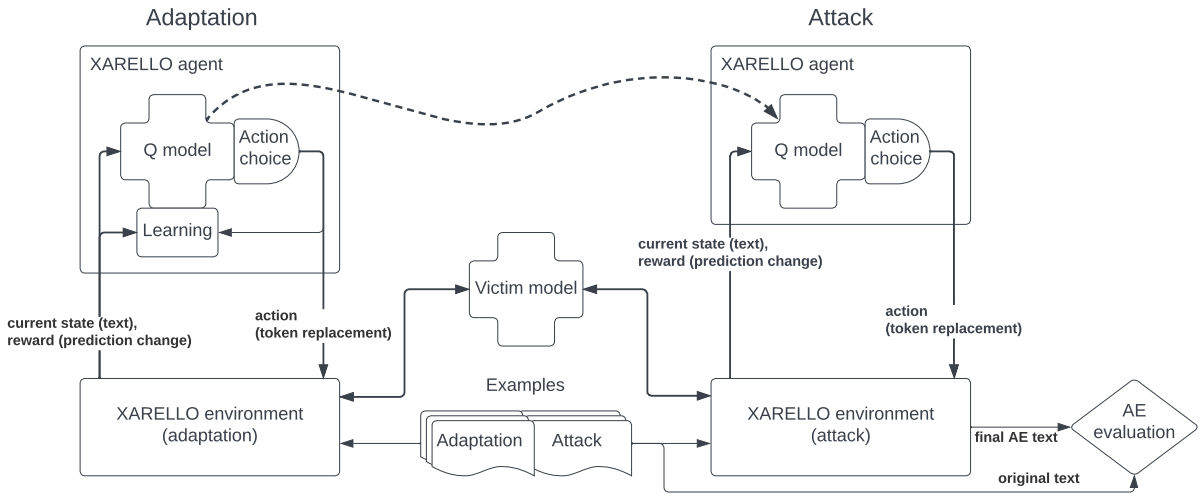

Figure 1: Conceptual schema of the XARELLO elements in the adaptation and attack phase.

art, both in terms of the more subtle modifications and lower number of attempts necessary. The victims, against which our attacker is tested, include a state-of-the-art LLM (GEMMA), which surprisingly appears the most vulnerable to the adaptive attack. We also qualitatively analyse the generated examples to better understand the techniques our models learn during the adaptation. The code for XARELLO is openly available to encourage research into AEs as well as building more robust classifiers[1].

2 Related work

The challenge of discovering AEs began in image classification research (Szegedy et al., 2013), where neural networks were discovered to change predictions after noise was added to the input. Generalising this approach to text is not trivial due to its discrete nature and the lack of 'imperceptible noise' equivalent, but several approaches emerged (Zhang et al., 2020). Typically, they rely on an iterative procedure of replacing fragments of input text with words that are similar in terms of meaning (Ren et al., 2019; Garg and Ramakrishnan, 2020; Li et al., 2020; Alzantot et al., 2018), in terms of visual appearance, or using character replacements (Gao et al., 2018). Recent work has been improving this paradigm (Liu et al., 2023) or abandoning it in favour of sentence-to-sentence paraphrasing, e.g. using auto-encoders (Li et al., 2023).

Misinformation detection is a scenario with a high probability of adversarial action. Several studies have been performed to assess the robustness of the two most-popular tasks: Fact-checking, usually using manually crafted rules (Zhou et al., 2019; Thorne et al., 2019; Hidey et al., 2020); and fake news detection (Jin et al., 2020; Ali et al., 2021; Brown et al., 2020; Smith et al., 2021). We also need to mention the novel threat of machine-generated text used for misinformation, and the models for its detection (Crothers et al., 2023) being vulnerable to attacks (Su et al., 2023).

In order to perform an evaluation of XARELLO in various scenarios, we rely on the previous systematic study of adversarial robustness in the credibility assessment context (Przybyła et al., 2023), taking into account four misinformation-detection tasks and two victim classifiers. This will allow us to compare our solution to the eight AE generators evaluated there.

Finally, a few attempts have already been made to use reinforcement learning (RL) in the context of AE generation. Our solution has certain similarities with that of Vijayaraghavan and Roy (2019), who also apply RL to find the most successful word substitutions, but in a less challenging setup: attacking a CNN network performing sentiment analysis and news classification. Other work involving RL include that of Li et al. (2021) and Chen et al. (2023). However, our study is the first to perform *adaptive* AE generation for the misinformation text, where a victim vulnerability model is first explicitly learned and then deployed for a more efficient attack.

3 Methods

XARELLO modifies given text not only based on the current input (original content), but also taking into account the outcome of previous attempts

[1] https://github.com/piotrmp/xarello

made against the same victim classifier. The whole process has two phases: *adaptation* and *attack*.

Figure 1 shows a schema of our solution. We map the problem of generating AEs (section 3.1) to the reinforcement learning paradigm through the *XARELLO environment* (section 3.2). During adaptation, a *XARELLO agent* (section 3.3) learns to perform actions (token replacements) that maximise its reward (change in the victim's prediction). The core of the model is a neural network estimating the outcome of making modifications to the input text. During the attack, the learned model, encoding information about the vulnerabilities of the victim, can be used to generate a multitude of adversarial examples, undergoing evaluation.

3.1 Preliminaries

We focus on binary text classification task using pairs (x_i, y_i), where x_i is a text fragment and y_i is a binary label denoting credibility of the text (section 4.3). The victim of the attack is a classifier f, which, for a given example x_i, provides a binary output label $f(x_i) \in \{0, 1\}$, but also probabilities of the positive class $f_p(x_i) \in (0, 1)$. The goal of the attack is to come up with a modification function m, such that the difference with the original example is small ($m(x_i) \approx x_i$), but the victim changes its decision ($f(x_i) \neq f(m(x_i))$, for example x_i = *Drinking orange juice causes DEATH!* and $m(x_i)$ = *Drinking orange juice provokes DEATH!*. Here we consider both the *targeted* scenario, taking into account only examples of non-credible text, for which the classifier made the correct decision ($y_i = f(x_i) = 1$); as well the *untargeted* one, where all examples are included.

3.2 XARELLO environment

The basic steps in our model are the same as in most methods for AE generation in text, i.e. sequential modifications, each consisting of replacing a word by a candidate from a pre-computed list, until the victim changes its decision (see section 2). Usually, no single replacement can result in an AE, but several are necessary. To learn an optimal strategy for such a task, we use the *reinforcement learning* (RL) framework (Sutton and Barto, 2018). We define the environment in the following way:

- an environment state s includes the following:
  - $x_{i,j}^{(t)}$ – the current form (in step t) of the i-th target text, expressed as a sequence of N tokens ($j \in \{1 \dots N\}$),
  - $f(x_i)$ – the decision of the victim for the original text.
- an *action* a made by an agent: a pair (j, k) including the positions of the changed token j and the replacement candidate z_k from a pre-computed list $z_1, z_2, \dots z_K$.
- a reward returned in response to an action:
  - 1, if the provided example is an AE,
  - -1, for an attempt to modify a non-word token (see section 4.5).
  - otherwise, $[f_p(x_i^{(t)}) - f_p(x_i^{(t-1)})] \times [1 - 2 \times f(x_i)]$, i.e. the difference in the score compared to previous state, computed with respect to the original class, so that positive values indicate the victim getting closer to changing the decision.

Adaptation: During adaptation, the environment presents subsequent examples to the agent. While it would be preferable to have only unique examples, the limited data size means that examples are repeated for several *epochs*. Since an agent is unlikely to find an AE by just a single word replacement, it is allowed several modifications (*steps*) until an AE is successful or the maximum number of steps ($M_S = 5$) is achieved. For example, an agent might try *Drinking orange juice provokes DEATH!*, then *Consuming orange juice provokes DEATH!*, then *Consuming orange juice brings DEATH!*, and so on. Such a sequence, called *episode*, is attempted $M_E = 5$ times (with text reset to the originals state in between) before the next example is used. We encourage variability of actions between episodes through the penalisation of action reuse (section 4.5).

Attack: In the attack stage the Q model is frozen and no learning is performed, allowing more elaborate action sequences as follows:

1. 10 episodes of up to 5 steps,
2. 5 episodes of up to 10 steps,
3. 2 episodes of up to 25 steps,
4. 1 episode of up to 50 steps.

Performing several episodes for the same number of steps allows the attacker to makes several attempts to create an AE with few changes, before performing deeper modifications. As during training, the text is reset to the original form between episodes and penalisation is used to encourage variation between attempts (section 4.5). For longer input text (news bias and rumour detection tasks,

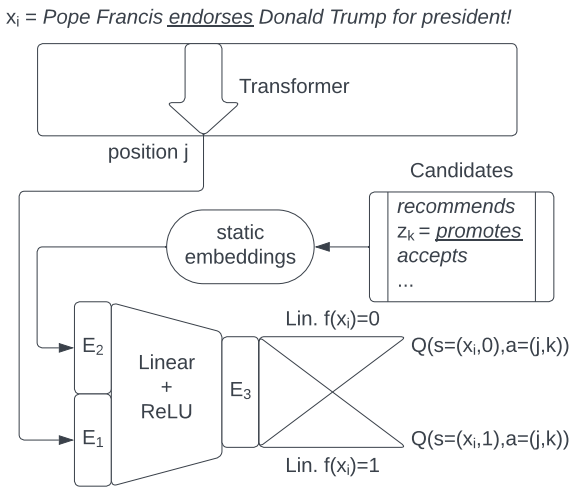

Figure 2: Neural network used as Q model.

see section 4.3), the number of steps allowed is multiplied by 5. The process can stop at any point if an AE for the current text is found, which is sent for evaluation.

3.3 XARELLO agent

The implementation of the XARELLO agent is based on Q-learning (Watkins, 1989), which involves estimating the value of $Q(s, a)$, i.e. the expected reward achieved from making action a in state s and following a greedy strategy. In particular, we implement a *deep Q-network* (François-Lavet et al., 2018), where the estimation is produced by a deep neural network, subsequently trained based on the actually observed rewards.

Q model: We compute the value of $Q(s = (x_{i,j}^{(t)}, f(x_i)), a = (j, k))$ as follows (see Fig. 2):

1. For each token position j, its E_1-dimensional embedding is computed through a Transformer (Vaswani et al., 2017) encoder working on the current text $x_{i,j}^{(t)}$,
2. It is concatenated with a pre-computed E_2-dimensional embedding of candidate z_k, forming a $E_1 + E_2$-dimensional representation of each possible action (j, k),
3. A linear layer with rectified linear activation is applied, reducing the dimensionality to E_3,
4. Depending on the value of original prediction $f(x_i)$, one of two independent final linear layers is used, reducing the dimensionality to a scalar, containing the value of $Q(s, a)$.

The neural network is implemented so that it computes the Q value for every possible action in a given state in a single execution.

Action choice: Choosing an action based on the Q value depends on the phase. In attack, simply the action with maximal Q value is selected (*greedy* strategy). In the adaptation phase, a random action may also be made with the probability equal exploration factor $\epsilon \in [0, 1]$ – an ϵ-greedy strategy (Sutton and Barto, 2018). Further information including parameter values and underlying components is in section 4.5.

Learning: As usual in fitted Q-learning, after an action is performed, the value of Q estimation is compared with the observed reward and discounted expected reward (using discount coefficient γ) and the resulting discrepancy is used as a loss for training the underlying neural network.

4 Evaluation

Since our solution is motivated by the adversarial scenarios in the misinformation space, we base our evaluation on the BODEGA framework (Przybyła et al., 2023), which is designed specifically for this area. It enables the evaluation in four misinformation detection tasks: style-based news bias assessment (HN), propaganda detection (PR), fact checking (FC), rumour detection (RD), all for English. A non-credible (positive, label=1) example, which should be detected by a classifier, is a news item from a hyper-partisan source, a sentence including a propaganda technique, a fact refuted by the provided evidence, or a thread initiated by a rumour. Examples are shown in table 4 in appendix E. All of the tasks are based on data released on CC licences (Potthast et al., 2018; da San Martino et al., 2020; Thorne et al., 2018; Han et al., 2019).

BODEGA enables an evaluation of attacks on two classifiers, based on BiLSTM (Hochreiter and Schmidhuber, 1997) and fine-tuned BERT (Devlin et al., 2018). Additionally, in order to understand the vulnerability of the modern LLMs, we test against 2-billion-parameter GEMMA (Gemma Team and Google DeepMind, 2024).

4.1 Performance measures

The attack performance is assessed by comparing each original examples with the produced AE and computing four measures:

1. *confusion score*: 1 if the example provided is a successful, 0 otherwise,
2. *semantic score*: a measure of the meaning preservation between the original text and the

Task	Adaptation train	Adaptation eval	Attack	Positive %
HN	3,200	400	400	50.00%
PR	2,920	400	416	29.42%
FC	3,200	400	405	51.27%
RD	1,670	400	415	32.68%

Table 1: The division of the BODEGA datasets for the purpose of adaptation and final attack with the percentage of positive (non-credible) instances.

AE, computed using BLEURT (Sellam et al., 2020) and clipped to (0,1),

3. *character score*: a measure of character-level changes, computed using Levenshtein distance (Levenshtein, 1966) and scaled as a similarity score in (0,1),
4. *BODEGA score*: a product of the above.

These quantities are averaged over all examples in a given experiment. More information on these measures, including the handling of multi-sentence inputs, could be found in the BODEGA framework (Przybyła et al., 2023). Additionally, we record the average number of queries a method needs to perform on the victim classifier before an AE is generated, as a measure of how realistic a given strategy is to be used in practice.

During the adaptation phase, we measure its progress through certain indicators after each epoch, both on the training data and a held-out development set (used greedily). These include mean reward value, the fraction of the episodes that end with a success, and the number of steps involving a given text before an AE is found.

4.2 Qualitative analysis

In addition, we provide a qualitative analysis of AEs generated by the XARELLO system against the BERT classifier in the targeted PR task. In Section 6, we make some observations on linguistic patterns that appear in this subset of AEs. Human evaluation is especially important for NLP models that generate text which people may read, or use in text generation to aid replicability (Belz et al., 2023). These models must also generate naturalistic text which reflects qualities such as grammaticality, fluency, and coherence (van der Lee et al., 2021) in order to be usable in practice, i.e. as misinformation content.

4.3 Data

Table 1 shows the data distribution, based on the BODEGA framework. We do not use the data reserved for victim training (not included in the table) and leave final attack portion unchanged, enabling comparison with previous work. We employ the development subset in XARELLO, splitting it into adaptation-train (for Q adaptation) and adaptation-eval (for monitoring the process, see measures above). We also show what fraction of each dataset as a whole is positive, i.e. non-credible.

4.4 Experiments

Each experiment starts with performing the adaptation for 20 epochs. During every epoch, firstly the adaptation-train data are used to learn from the experiences and update the network accordingly. Afterwards, the held-out adaptation-eval portion is used (with the greedy strategy and no weight updates) to measure the adaptation performance.

After the adaptation is finished, the model that performed the best on adaptation-eval, i.e. needed the least steps on average to reach an AE, is selected for final attack evaluation. This is performed by connecting the learned Q model to an environment working in attack mode and evaluating the quality of the AEs with BODEGA.

In total, 12 adaptation processes are performed (against three victim classifiers for each of the four tasks), which are followed by two evaluation scenarios: targeted or untargeted. We compare an adapted XARELLO against:

- BERT-ATTACK (Li et al., 2020), performing a procedure of iterative replacement of words by candidates from a language model, fairly similar to XARELLO, but without any adaptation to the victim. BERT-ATTACK achieved the best result among those evaluated on BODEGA (Przybyła et al., 2023).
- DeepWordBug (Gao et al., 2018), a simpler approach, replacing individual characters in the selected words, aiming to preserve visual similarity to the original text. DeepWordBug was also the best-performing in some attack scenarios in BODEGA.
- XARELLO-raw, a version of the XARELLO agent which was not adapted to the victim. Testing this version allows us to make sure the observed differences are due to adaptation process, rather than the attack procedure.

4.5 Optimisation details

Preprocessing: The maximum length of a text fragment is $N = 512$ tokens and all instances are padded accordingly. For each text and each

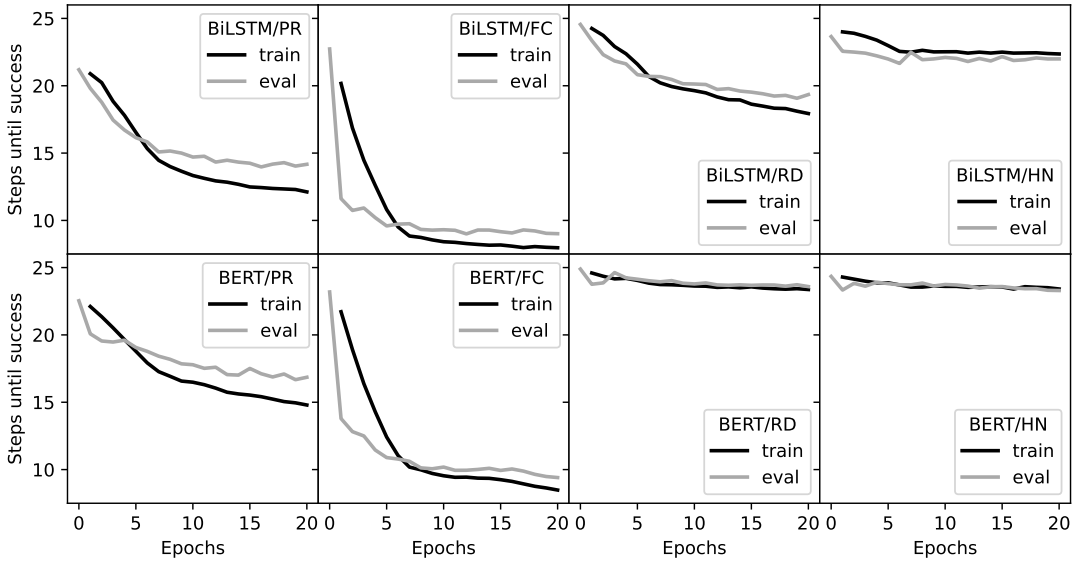

Figure 3: Improvement of the XARELLO attackers during the adaptation process, illustrated using the average number of steps until an AE is found, shown for the data used in training (*train*) and a held-out portion (*eval*), for each epoch. Shown for each of the tasks and victims: BiLSTM (upper row) and BERT (lower row).

non-padding token, the $K = 20$ replacement candidates are obtained by applying language modelling through BERT (Devlin et al., 2018) in `bert-base-cased` variant, implemented in *HuggingFace Transformers* (Wolf et al., 2020). No masking is used, as in BERT-ATTACK (Li et al., 2020), and the most likely tokens for each position are treated as candidates, disregarding the original word and special tokens.

Neural network: We use BERT (configured as above), to obtain embeddings of size $E_1 = 768$. To represent candidates, we use static *fastText* (Mikolov et al., 2017) vectors, i.e. the `facebook/fasttext-en-vectors` model from *HuggingFace*, returning an embedding of size $E_2 = 300$. The reduced representation has size $E_3 = 8$. The Q network includes the 110 million parameters of BERT and 8570 in the further layers.

Q override: In order to indicate that non-word tokens (`[CLS]`, `[SEP]` or `[PAD]`) cannot be changed, the reward for attempting to replace them is set to -1. Moreover, the Q value obtained from the neural network is overridden using two rules: (1) the value for replacing special tokens are set to -1 and (2) the value for actions that have already been applied for this text in the current sequence of episodes are reduced by a factor of -0.1. This penalisation mechanism makes it possible to generate diverse actions even when Q network remains unchanged, esp. in the attack phase. Both of the alterations correspond to behaviours that are beneficial for the rewards and would be learnt eventually, but introducing them accelerates the adaptation.

Further details on hardware and computing times, software implementation, adaptation process and parameter tuning can be found in appendix B.

5 Results

Figure 3 shows the progress made during the adaptation to the BiLSTM and BERT victims (the results for GEMMA are included as figure 4 in appendix A). We plot the average number of steps made until an AE is found or the limit is reached, taking values between 5 (the AE is found on first try in all 5 episodes) and 25 (all 5 steps in the 5 episodes are used). All models start with a value close to the maximum and manage to improve over time, but the gains are more pronounced for the PR and FC tasks than RD or HN. This is understandable, as the text fragments involved in the latter two (news articles and rumour threads) are much longer, so it is relatively rare to see an AE generated within the 5 modifications allowed during adaptation. We can also see that the BiLSTM victim, as a weaker classifier, is easier to attack, allowing an AE to be found in fewer steps after the adaptation.

It is encouraging to notice that the performance on the unseen eval dataset improves similarly, indicating that the model indeed learns vulnerabilities of the victim model instead of memorising the steps that prove successful for the training data. Towards the end of the 20-epoch process we see the im-

		Victim: BiLSTM				Victim: BERT				Victim: GEMMA			
				XARELLO				XARELLO				XARELLO	
	Measure	DWB	B-A	raw	full	DWB	B-A	raw	full	DWB	B-A	raw	full
PR	BODEGA	0.292	0.527	0.466	**0.632**	0.278	0.429	0.360	**0.512**	0.143	0.460	0.474	**0.697**
	conf.	0.382	0.800	0.928	0.990	0.363	0.697	0.769	0.962	0.190	0.724	0.899	0.986
	sem.	0.795	0.716	0.595	0.698	0.794	0.678	0.562	0.606	0.786	0.695	0.605	0.748
	char.	0.960	0.914	0.791	0.884	0.962	0.902	0.772	0.834	0.958	0.906	0.813	0.920
	queries	27.4	61.4	61.4	**15.0**	**27.4**	80.2	89.8	30.2	27.3	77.5	59.5	**14.9**
FC	BODEGA	0.484	0.598	0.640	**0.817**	0.440	0.535	0.559	**0.773**	0.074	0.566	0.577	**0.775**
	conf.	0.575	0.857	0.938	1.000	0.531	0.770	0.862	0.995	0.091	0.832	0.904	0.995
	sem.	0.855	0.728	0.733	0.837	0.843	0.726	0.708	0.800	0.829	0.718	0.698	0.802
	char.	0.984	0.954	0.917	0.975	0.982	0.953	0.902	0.970	0.983	0.939	0.902	0.969
	queries	54.4	132.8	56.0	**5.0**	54.3	146.7	74.1	**7.4**	53.9	192.2	66.3	**7.3**
RD	BODEGA	0.164	0.292	0.244	**0.650**	0.159	0.181	0.145	**0.227**	0.104	0.300	0.228	**0.314**
	conf.	0.243	0.790	0.537	0.973	0.229	0.439	0.333	0.436	0.152	0.725	0.434	0.492
	sem.	0.682	0.409	0.514	0.694	0.701	0.429	0.500	0.580	0.694	0.433	0.590	0.678
	char.	0.991	0.890	0.842	0.957	0.991	0.961	0.830	0.870	0.991	0.951	0.865	0.934
	queries	232.8	985.5	617.8	**84.0**	**232.7**	774.3	763.5	631.7	**239.0**	703.1	665.7	538.9
HN	BODEGA	0.406	**0.636**	0.496	0.612	0.223	**0.601**	0.340	0.341	0.240	**0.546**	0.485	0.528
	conf.	0.527	0.980	0.760	0.848	0.287	0.965	0.560	0.583	0.307	0.905	0.752	0.757
	sem.	0.771	0.656	0.689	0.737	0.777	0.638	0.644	0.607	0.783	0.622	0.676	0.715
	char.	0.998	0.988	0.933	0.975	0.998	0.972	0.918	0.937	0.998	0.965	0.930	0.963
	queries	396.2	487.9	445.7	**256.1**	**395.9**	648.4	599.8	564.4	385.9	943.0	427.7	**373.6**
Avg:	BODEGA	0.337	0.513	0.461	**0.678**	0.275	0.436	0.351	**0.463**	0.141	0.468	0.441	**0.578**
	queries	177.7	416.9	295.2	**90.0**	**177.6**	412.4	381.8	308.4	**176.5**	478.9	304.8	233.7

Table 2: Results of the evaluation of the XARELLO attacker on different datasets (PR, FC, RD and HN) in the untargeted scenario, measured according to BODEGA score, confusion score, semantic similarity score, character similarity score and average number of queriues. The performance of the adapted XARELLO (*full*) is compared to the attacker without adaptation (*raw*) and two separate approaches: DeepWordBug (*DWB*) and BERT-ATTACK (*B-A*). The best values of BODEGA score and the lowest numbers of queries in each combination are highlighted.

provements on the eval dataset slow down, suggesting that further training would result in overfitting, which confirms the preliminary experiments with 50 epochs (see appendix B).

Table 2 shows the results of the main experiment in untargeted scenario (with all data), carried out by taking a Q neural network optimised during adaptation and applying to the attack data portion. The performance indicators averaged over all scenarios (final rows) confirm the benefits of the proposed approach: it achieves better-quality AEs, reflected with a higher BODEGA score. The gains are most pronounced against the BiLSTM victim, where XARELLO achieves the score of 68%, compared to 51% of BERT-ATTACK, needing only 90 queries instead of 417. We also see an improvement over baseline in case of BERT, but it is interesting to notice that GEMMA, the model of largest size and best classification performance, is quite vulnerable against XARELLO attacks (58% compared to 47% of BERT-ATTACK).

Overall, DeepWordBug produces examples that are semantically and visually similar to the original, but achieve success only in some cases. For example, in BiLSTM fact-checking scenario, DeepWordBug has a confusion score of 57%, BERT-ATTACK of 86%, but XARELLO reaches 100%. This is possible due to the adaptation process, as XARELLO raw ranks similarly to BERT-ATTACK and only the full version achieves the improvements.

The performance differs across tasks: XARELLO shows improvement in all of them except news bias assessment, especially against the BERT victim. This is most likely due to the length of the input: news articles often fill the whole 512-token window, resulting in 512*20 possible actions – a space unlikely to be thoroughly explored within the limits of the adaptation. The quality of the sample AEs remains high, but they are just not found for as many examples as in BERT-ATTACK. This is in line with the slow adaptation for this combination visible in Figure 3 (BERT/HN) and research showing fake news detection as relatively robust (Jin et al., 2020).

On the other hand, the performance gains for tasks with shorter text are substantial. In evaluation against fact-checking task XARELLO not only beats BERT-ATTACK in terms of BODEGA score (77% vs 53%), but is able to reach an AE in 7.42 queries on average, rather than 146.

The results for the targeted attacks are shown in table 3 in appendix D. The general outlook is very similar, but the targeted attacks appear more successful, especially against BERT and GEMMA.

6 Linguistic analysis

In order to see what these improved performance metrics look like in actual output utterances, we perform a textual analysis on 67 AEs against BERT, generated by XARELLO from the PR task in the targeted scenario. These are examples where low-credibility text was recognised as such by the victim model, but the modifications introduced by XARELLO changed this decision. Examples of the described phenomena are shown in appendix C.

Our main takeaway is that the XARELLO agent strongly relies on making replacements at the *sub-word* level. Some of these render clear non-words which result in sentences becoming completely ungrammatical. Other non-words may pose less of a problem to reader, as they are typographically very similar to the original text. A similar phenomenon occurs in generated non-words which may appear to be infrequent or archaic words which match the orthographic and phonological rules of English[2].

It is possible that readers may not notice these spelling mistakes. In multiple studies over decades, the first and last letters in a word contribute more strongly to recognition (Huebert and Cleary, 2022), for example when “hypocritically" is replaced by “*hypoclipically". AEs may therefore be ungrammatical, but still effective.

There are also patterns of adjectival replacement which appear to perform a form of semantic bleaching, or that introduce euphemistic language by replacing an emotionally charged noun or noun phrase with a pronoun[3] or a more generalistic noun[4]. This strategy is not always successful, with around half of this type of replacement resulting in ungrammatical utterances[5]. Moreover, the agent may be too greedy and remove crucial constituents of an utterance[6]. We also discovered words which XARELLO has learned to retain, or avoid and provide replacements. It often chooses “new" or “big" to replace more semantically-transparent or emotive words, and this links to our observations about adjectival and pronominal replacement.

The observed modification types may stem from the nature of XARELLO’s victim BERT’s subword tokenisation method, as well as our use of fastText to represent word replacement candidates. In order to ‘fool’ the classifier, XARELLO may rely too strongly on replacing pieces of words whose output resembles the orthographic and morphological rules of English but which may not be acceptable to real-world readers.

Possible methods to mitigate ungrammatical output could be to check output tokens against the N-gram probability of the AE, using semantic similarity as a heuristic for whole-token replacement, penalising tokens which do not appear in a lookup lexicon, or by using reinforcement learning from human feedback (Ziegler et al., 2020).

7 Limitations

Despite showing positive results, our study has several limitations. Firstly, in casting the AE generation as a RL problem (section 3.2), we discard the possibility of adding new words to the original text, which is possible in some previous AE generators, such as BAE (Garg and Ramakrishnan, 2020). Word deletion is not allowed either, even though it is one of the most natural ways of changing the form of the text while preserving its meaning (Shardlow and Przybyła, 2023). Finally, we do not perform any special treatment of subwords, e.g. as in BERT-ATTACK (Li et al., 2020). These operations are excluded in order to reduce the size of the action space, but incorporating them would be a promising avenue for future research.

Secondly, due to the long processing time, we performed only a basic exploration of the influence of the many parameters present in our solution (see appendix B). Some of these, e.g. discounting coefficient, do not have an obvious meaning in context of AE search, and their best value could only be discovered through systematic tuning. Others, such as dimensions of the Q network, number of steps and episodes, likely depend on a particular task, so would have to be tuned for each of them separately. Finally, some, such as number of candidates, would almost certainly improve the performance, but at the cost of longer adaptation time. However, these experiments might be justified if we want to simulate an attacker that consistently operates against a specific target.

Moreover, classifiers more elaborate than included here could be tested as victim models as well. We decided to use BiLSTM and BERT in the interest of comparability with previous solutions, numerous of which were evaluated against BODEGA (Przybyła et al., 2023), and GEMMA to illustrate vulnerability of modern LLMs. It is

[2]Original: “lives and **vocations**"→AE: “***vassations**"
[3]Original: “his **aggressive** behaviour"→AE: “**own**"
[4]Original: “that type of **injustice**→AE: “**work**".
[5]Original:“from the **american people**"→AE: “**my us**"
[6]Original:“reported on a **gaping hole** in"→AE: “*"

interesting to notice that the latest of the tested approaches is also the most prone to attacks. Future work might verify if this is caused by reasons connected to our setup, e.g. relatively small datasets for tuning a network of this size, or a more fundamental weakness of very large models.

Even though misinformation is an equally grave problem for non-English Internet, our solution is only evaluated on English datasets. However, XARELLO does not depend on English in any particular way and could be applied to any language, as long as a Transformer model for it exists.

Finally, the results on the news bias assessment indicate our approach does not generalise very well to the case when numerous changes in a long text need to be made. This is because the final reward typically could not be achieved within the short horizon of the adaptation episodes. A more exhaustive search for solutions should happen during adaptation in such cases, including attacks of increasing length, as in the attack phase.

8 Ethical impact

The work in the domain of adversarial robustness needs to be scrutinised to make sure it does not aid the malicious actors. However, discovering AEs is definitely more likely to help build up the defences. Firstly, the examples we generate cannot be used directly to perform any attacks. That is because AEs are not transferable, so they would work only with the models they were discovered for, i.e. the victim classifiers. The models used for content moderation are likely trained using newer architectures and proprietary internal data. Secondly, despite the progress in the domain, most attack scenarios still require dozens or even hundreds of attempts are impossible to conduct in practice.

More generally, the AEs are vulnerabilities that exist due to the nature of neural networks and research such as ours is only revealing, not creating them. In our view, it is better that such techniques are obtained and discussed within the transparent research discourse rather than they would be discovered just by misinformation spreaders. For these reasons, we have decided to make the XARELLO code available[7].

9 Conclusion

To sum up, XARELLO adapts well to the weaknesses of a victim model and in all scenarios, except with very long text, achieves superior performance. This result applies to various victim models, from small RNN networks to classifiers based on large modern fine-tuned LLMs. This allows us not only to find AEs for more examples, and of better quality, but also do this with fewer attempts. The evaluation becomes more realistic, as it is more likely that a platform would allow a user to send 5 consecutive messages of similar content to find an AE, rather than 133, needed by other methods.

We rely on an expectation that an attacker has already some experience with the current classifier. This is a much lower bar than in *white-box* attacks, assuming complete access to victim model weights. Nevertheless, in practice it will depend on the internal operations both of misinformation spreaders (e.g. experience retention) and content platforms (e.g. model updating frequency).

Ultimately, AEs allow us to find and understand the weaknesses of the investigated models before they are deployed. We can build on these methods to improve the model robustness. Our contribution could be easily used for this purpose, i.e. by including the generated AEs in the training data, as in the *adversarial training* paradigm (Bai et al., 2021).

We hope that by making the code of XARELLO openly available, we enable such use-cases and contribute to more reliable role of automatic classifiers in making the Internet safer.

Acknowledgements

The work of P. Przybyła is part of the ERINIA project, which has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101060930. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the funders. Neither the European Union nor the granting authority can be held responsible for them. We gratefully acknowledge Polish high-performance computing infrastructure PLGrid (HPC Centers: ACK Cyfronet AGH) for providing computer facilities and support within computational grant no. PLG/2023/016896. We also acknowledge support from Departament de Recerca i Universitats de la Generalitat de Catalunya (ajuts SGR-Cat 2021) and from Maria de Maeztu Units of Excellence Programme CEX2021-001195-M, funded by MCIN/AEI /10.13039/501100011033.

[7] https://github.com/piotrmp/xarello

References

Hassan Ali, Muhammad Suleman Khan, Amer AlGhadhban, Meshari Alazmi, Ahmad Alzamil, Khaled Alutaibi, and Junaid Qadir. 2021. All Your Fake Detector are Belong to Us: Evaluating Adversarial Robustness of Fake-News Detectors Under Black-Box Settings. *IEEE Access*, 9:81678–81692.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating Natural Language Adversarial Examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2890–2896, Brussels, Belgium. Association for Computational Linguistics.

Tao Bai, Jinqi Luo, Jun Zhao, Bihan Wen, and Qian Wang. 2021. Recent Advances in Adversarial Training for Adversarial Robustness. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 4312–4321. ijcai.org.

Anya Belz, Craig Thomson, Ehud Reiter, and Simon Mille. 2023. Non-repeatable experiments and non-reproducible results: The reproducibility crisis in human evaluation in NLP. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3676–3687, Toronto, Canada. Association for Computational Linguistics.

Brandon Brown, Alexicia Richardson, Marcellus Smith, Gerry Dozier, and Michael C. King. 2020. The Adversarial UFP/UFN Attack: A New Threat to ML-based Fake News Detection Systems? In *2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020*, pages 1523–1527. IEEE.

Kuan-Chun Chen, Chih-Yao Chen, and Cheng-Te Li. 2023. ANTI-Disinformation: An Adversarial Attack and Defense Network Towards Improved Robustness for Disinformation Detection on Social Media. In *2023 IEEE International Conference on Big Data (BigData)*, pages 5476–5484. IEEE.

Evan N. Crothers, Nathalie Japkowicz, and Herna L. Viktor. 2023. Machine-Generated Text: A Comprehensive Survey of Threat Models and Detection Methods. *IEEE Access*, 11:70977–71002.

Giovanni da San Martino, Alberto Barrón-Cedeño, Henning Wachsmuth, Rostislav Petrov, and Preslav Nakov. 2020. SemEval-2020 Task 11: Detection of Propaganda Techniques in News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation (SemEval-2020)*, pages 1377–1414.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. 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*, pages 4171–4186. Association for Computational Linguistics.

Renee DiResta, Kris Shaffer, Becky Ruppel, David Sullivan, Robert Matney, Ryan Fox, Jonathan Albright, and Ben Johnson. 2019. The Tactics & Tropes of the Internet Research Agency. Technical report, Congress of The United States.

Vincent François-Lavet, Peter Henderson, Riashat Islam, Marc G Bellemare, and Joelle Pineau. 2018. An Introduction to Deep Reinforcement Learning. *Foundations and Trends in Machine Learning*, 11(3-4):219–354.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *Proceedings - 2018 IEEE Symposium on Security and Privacy Workshops, SPW 2018*, pages 50–56. IEEE.

Siddhant Garg and Goutham Ramakrishnan. 2020. BAE: BERT-based Adversarial Examples for Text Classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6174–6181, Online. Association for Computational Linguistics.

Gemma Team and Google DeepMind. 2024. Gemma: Open Models Based on Gemini Research and Technology. Technical report, Google DeepMind.

Sooji Han, Jie Gao, and Fabio Ciravegna. 2019. Neural language model based training data augmentation for weakly supervised early rumor detection. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2019*, pages 105–112. Association for Computing Machinery, Inc.

Christopher Hidey, Tuhin Chakrabarty, Tariq Alhindi, Siddharth Varia, Kriste Krstovski, Mona Diab, and Smaranda Muresan. 2020. DeSePtion: Dual Sequence Prediction and Adversarial Examples for Improved Fact-Checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8593–8606. Association for Computational Linguistics (ACL).

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Andrew M Huebert and Anne M Cleary. 2022. Do first and last letters carry more weight in the mechanism behind word familiarity? *Psychonomic Bulletin & Review*, 29(5):1938–1945.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is BERT Really Robust? A Strong Baseline for Natural Language Attack on Text Classification and Entailment. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020,*, pages 8018–8025. AAAI Press.

Diederik P. Kingma and Jimmy Lei Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, San Diego, USA. ICLR.

Vladimir Iosifovich Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10:707–710.

Ang Li, Fangyuan Zhang, Shuangjiao Li, Tianhua Chen, Pan Su, and Hongtao Wang. 2023. Efficiently generating sentence-level textual adversarial examples with Seq2seq Stacked Auto-Encoder. *Expert Systems with Applications*, 213:119170.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. BERT-ATTACK: Adversarial Attack Against BERT Using BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6193–6202. Association for Computational Linguistics.

Yue Li, Pengjian Xu, Qing Ruan, and Wusheng Xu. 2021. Text Adversarial Examples Generation and Defense Based on Reinforcement Learning. *Tehnički vjesnik*, 28(4):1306–1314.

Han Liu, Zhi Xu, Xiaotong Zhang, Xiaoming Xu, Feng Zhang, Fenglong Ma, Hongyang Chen, Hong Yu, and Xianchao Zhang. 2023. SSPAttack: A Simple and Sweet Paradigm for Black-Box Hard-Label Textual Adversarial Attack. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13228–13235. AAAI Press.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2017. Advances in Pre-Training Distributed Word Representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. 2015. Human-level control through deep reinforcement learning. *Nature 2015 518:7540*, 518(7540):529–533.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. 2018. A Stylometric Inquiry into Hyperpartisan and Fake News. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 231–240. Association for Computational Linguistics.

Piotr Przybyła, Alexander Shvets, and Horacio Saggion. 2023. Verifying the Robustness of Automatic Credibility Assessment. *arXiv preprint arXiv:2303.08032*.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating Natural Language Adversarial Examples through Probability Weighted Word Saliency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1085–1097, Florence, Italy. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning Robust Metrics for Text Generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Matthew Shardlow and Piotr Przybyła. 2023. Simplification by Lexical Deletion. In *Proceedings of the Second Workshop on Text Simplification, Accessibility and Readability*, pages 44–50, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Mohit Singhal, Chen Ling, Pujan Paudel, Poojitha Thota, Nihal Kumarswamy, Gianluca Stringhini, and Shirin Nilizadeh. 2022. SoK: Content Moderation in Social Media, from Guidelines to Enforcement, and Research to Practice. In *The 8th IEEE European Symposium on Security and Privacy (EuroS&P 2023)*. IEEE.

Marcellus Smith, Brandon Brown, Gerry Dozier, and Michael King. 2021. Mitigating Attacks on Fake News Detection Systems using Genetic-Based Adversarial Training. In *2021 IEEE Congress on Evolutionary Computation, CEC 2021 - Proceedings*, pages 1265–1271. IEEE.

Pengcheng Su, Rongxin Tu, Hongmei Liu, Yue Qing, and Xiangui Kang. 2023. Adversarial Attacks on Generated Text Detectors. In *Proceedings - IEEE International Conference on Multimedia and Expo*, volume 2023-July, pages 2849–2854. IEEE Computer Society.

Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. *arXiv: 1312.6199*.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2019. Evaluating adversarial attacks against multiple fact verification systems. In *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International*

Joint Conference on Natural Language Processing, Proceedings of the Conference, pages 2944–2953. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. 2018. The Fact Extraction and VERification (FEVER) Shared Task. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*.

Mark Towers, Jordan K Terry, Ariel Kwiatkowski, John U Balis, Gianluca de Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Arjun KG, Markus Krimmel, Rodrigo Perez-Vicente, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Andrew Tan Jin Shen, and Omar G Younis. 2023. Gymnasium.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, and Emiel Krahmer. 2021. Human evaluation of automatically generated text: Current trends and best practice guidelines. *Computer Speech Language*, 67:101151.

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 *Advances in Neural Information Processing Systems*, volume 2017-Decem, pages 5999–6009. Curran Associates, Inc.

Prashanth Vijayaraghavan and Deb Roy. 2019. Generating Black-Box Adversarial Examples for Text Classifiers Using a Deep Reinforced Model. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2019, Würzburg, Germany, September 16-20, 2019, Proceedings, Part II*, volume 11907 of *Lecture Notes in Computer Science*, pages 711–726. Springer.

C.J.C.H. Watkins. 1989. *Learning from Delayed Rewards*. Ph.D. thesis, University of Cambridge.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Wei Emma Zhang, Quan Z. Sheng, Ahoud Alhazmi, and Chenliang Li. 2020. Adversarial Attacks on Deep-learning Models in Natural Language Processing. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(3).

Zhixuan Zhou, Huankang Guan, Meghana Moorthy Bhat, and Justin Hsu. 2019. Fake News Detection via NLP is Vulnerable to Adversarial Attacks. In *ICAART 2019 - Proceedings of the 11th International Conference on Agents and Artificial Intelligence*, volume 2, pages 794–800. SciTePress.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2020. Fine-tuning language models from human preferences.

A Adaptation process for GEMMA

Figure 4 shows the adaptation process for the GEMMA victims.

B Implementation details

Software implementation: The Q-learning environment is defined in terms of `Env` class in the *gymnasium* framework for RL (Towers et al., 2023). The neural network is implemented in *pyTorch* (Paszke et al., 2019).

Performance: The adaptation process is executed on a machine using one NVIDIA A100 GPU with 40 GB RAM. The duration of the process (all 20 epochs) varies depending on the victim and task performed, taking from 18 hours (BiLSTM, PR) to 42 hours (BERT, HN).

Parameter tuning: Due to the length of the adaptation process, only very limited parameter tuning was performed. To reduce the necessary processing time, in all of these experiments we used a smaller version of the model (for input up to 128 tokens), 1500 instances for adaptation and 400 for testing, both within the development portion of the PR task. Each run took around 10 hours to complete, save for the adaptation length experiment, taking proportionally longer. We tested separately adaptation length (20 or 50 epochs), memory mechanism (experiences either previously observed or drawn from memory of 4000), warmup periods (0.1, 0.3, 0.5, 0.7. 0.9), discounting parameter (0.0 or 0.5) and number of candidates (10 or 20). In general, our observations indicate low variability of the results within the ranges tested, but the best variants were selected for the main evaluation.

Adaptation: We train for 20 epochs on the adaptation dataset. The discounting coefficient is set to $\gamma = 0.5$ and exploration factor ϵ falls linearly during warmup period from 100% at the beginning of the process to 10% after 30% of the adaptation are finished and remains constant afterwards. As in the seminal work on deep RL (Mnih et al., 2015), we use a memory of previous experiences. Up to 4000 experiences are kept in a queue and 16 are randomly selected for Q update at every step. This learning is initiated every time 16 new experiences are added to the memory. The neural network is updated using Adam optimiser (Kingma and Ba, 2015) with a constant learning rate of 2×10^{-5}

C Qualitative analysis: examples

Changed characters by the agent are in boldface, and the star (*) symbol indicates incoherence, ungrammaticality or disfluency.

Examples where subwords are replaced rendering an ungrammatical sentence:

- Original: *"...doctors are warning that it will be continuing to s**pread** and **worse**n"*
- AE: *"*...doctors are warning that it will be continuing to s**low** and **bad**n"*
- Original: *"is already **reel**ing over the revelations...a Cardinal over weekend, has been **c**redibly accused"*
- AE: *"*is already **poised**ing over the revelations...a Cardinal over the weekend, has been **now**redibly accused"*

Examples of non-words which are typographically similar:

- Originals: *"me**n**ace", "hypo**crit**ically", "**bl**atently", "co**ll**uded"*
- AEs: *"*meace", "*hypo**clip**ically", "*b**r**atently", "*co**po**luded"*

Example of a non-word which may appear like an infrequent or archaic word:

- Original: *"many who have spent their lives and v**oc**ations"*
- AE: *"*many who have spent their lives and v**ass**ations"*

Examples of adjectival replacement resulting in euphemistic language:

- Original: *"that type of **injustice**"*
- AE: *"that type of **work**"*

Examples of a pronoun replacing a noun/noun phrase:

- Original: *"his **aggressive** behaviour", "**vicious** comments", "treated as **criminals**"*
- AE: *"his **own** behaviour", "**his** comments", "treated as **it**"*

An example where this does not work well:

- Original: *"from the **american people**"*
- AE: *"*from the **my us**"*

An example where a whole constituent of a sentence is removed unsuccessfully:

- Original: *"reported on a **gaping hole** in"*
- AE: *"*reported on a in"*

D Results in the targeted attacks

Table 3 includes the results of the evaluation in the targeted scenario.

E Text examples

Table 4 shows examples of credible and non-credible text in each task.

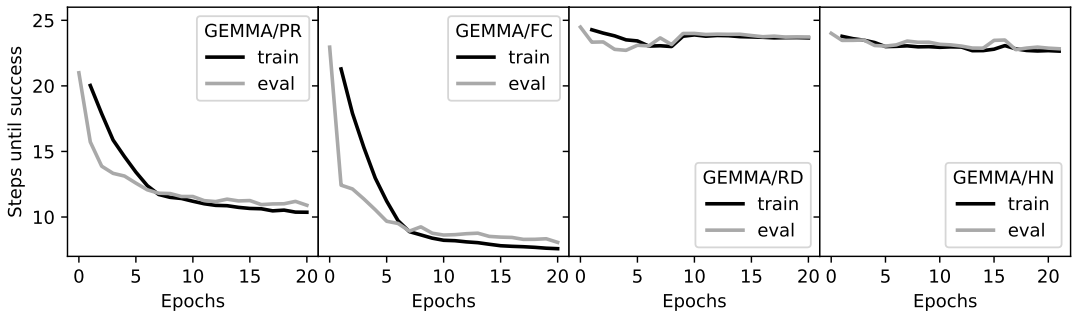

Figure 4: Improvement of the XARELLO attackers during the adaptation process, shown for each of the tasks and the GEMMA victims. See figure 3 and the main text for more information.

		Victim: BiLSTM				Victim: BERT				Victim: GEMMA			
				XARELLO				XARELLO				XARELLO	
	Measure	**DWB**	**B-A**	**raw**	**full**	**DWB**	**B-A**	**raw**	**full**	**DWB**	**B-A**	**raw**	**full**
PR	BODEGA	0.560	0.658	0.588	**0.682**	0.501	0.503	0.432	**0.523**	0.292	0.553	0.568	**0.617**
	conf.	0.720	0.940	0.980	1.000	0.640	0.787	0.760	0.907	0.378	0.851	0.905	0.986
	sem.	0.808	0.744	0.668	0.725	0.811	0.691	0.648	0.633	0.797	0.700	0.690	0.676
	char.	0.962	0.937	0.872	0.932	0.965	0.920	0.862	0.889	0.968	0.920	0.893	0.911
	queries	35.3	50.1	40.0	**10.3**	**36.0**	99.9	75.6	53.4	36.0	94.1	43.4	**23.3**
FC	BODEGA	0.540	0.594	0.613	**0.779**	0.224	0.413	0.471	**0.764**	0.063	0.496	0.513	**0.781**
	conf.	0.642	0.851	0.946	1.000	0.268	0.621	0.737	1.000	0.077	0.759	0.841	0.995
	sem.	0.854	0.726	0.706	0.803	0.847	0.708	0.700	0.789	0.836	0.701	0.676	0.810
	char.	0.984	0.956	0.907	0.969	0.983	0.932	0.897	0.966	0.984	0.919	0.880	0.967
	queries	50.7	123.2	57.1	**4.5**	52.3	207.2	100.8	**8.1**	52.0	254.2	91.5	**8.1**
RD	BODEGA	0.615	0.426	0.420	**0.636**	0.388	0.299	0.324	**0.433**	0.237	0.408	0.420	**0.604**
	conf.	0.907	0.947	0.947	1.000	0.560	0.690	0.770	0.880	0.346	0.933	0.923	1.000
	sem.	0.686	0.462	0.511	0.664	0.700	0.446	0.491	0.559	0.693	0.455	0.521	0.649
	char.	0.988	0.975	0.838	0.952	0.990	0.971	0.812	0.856	0.989	0.961	0.839	0.919
	queries	153.6	130.6	224.4	**5.6**	**174.0**	366.1	422.6	222.5	161.9	259.4	297.0	**46.1**
HN	BODEGA	0.366	**0.613**	0.368	0.545	0.153	**0.567**	0.175	0.247	0.267	**0.575**	0.494	0.534
	conf.	0.473	0.958	0.599	0.820	0.198	0.948	0.314	0.465	0.342	0.947	0.797	0.775
	sem.	0.775	0.648	0.658	0.682	0.776	0.620	0.610	0.558	0.782	0.624	0.653	0.708
	char.	0.998	0.985	0.918	0.966	0.997	0.962	0.885	0.916	0.998	0.970	0.925	0.963
	queries	379.2	565.0	585.4	**316.2**	**389.8**	753.9	795.7	691.0	380.6	761.5	408.3	**366.4**
Avg:	BODEGA	0.520	0.573	0.497	**0.660**	0.317	0.445	0.350	**0.492**	0.215	0.508	0.499	**0.634**
	queries	154.7	217.3	226.7	**84.2**	**163.0**	356.8	348.7	243.8	157.6	342.3	210.0	**111.0**

Table 3: Results of the evaluation of the XARELLO attacker on different datasets in the **targeted** scenario. See table 2 and the main text for further explanation.

Task	Credible example	Non-credible example
HN	Challenges in the Courts to Obamacare Certainly, as the new national health care changes get underway, there are going to be many challenges to it in the courts. These challenges will prove quite telling for the general public about the state of the health care reforms, and their legitimacy. In recent news, a Detroit Federal judge just upheld major elements of the health care overhaul law. U.S. District Judge George Steeh explained in his 20 page decision that not having health insurance is basically an active decision to pay out of pocket for health care. With this ruling, he supported the constitutionality of the health care reform law, particularly that part of it that indicates that individuals need to have health coverage. (...)	Texas Board Of Education Approves Resolution To Limit Islam References Associated Press AUSTIN, Texas — The Texas State Board of Education adopted a resolution Friday that seeks to curtail references to Islam in Texas textbooks, as social conservative board members warned of what they describe as a creeping Middle Eastern influence in the nation's publishing industry. The board approved the one-page nonbinding resolution, which urges textbook publishers to limit what they print about Islam in world history books, by a 7-5 vote. Critics say it's another example of the ideological board trying to politicize public education in the Lone Star State. (...)
PR	The Italian Catholic daily La Nuova Bussola Quotidiana reports that not only did the pope see a letter from victims, but that the CDF, under Muller, "had already conducted an preliminary investigation into Barros and the other bishops close to Karadima which had led to the decision to relieve them of their duties."	Somehow the openly racist and anti-Semitic Farrakhan and his hateful organization have managed for decades to avoid being harshly denounced as such by the news media, which instead has spent the last two years attempting to smear Donald Trump as the new Hitler.
FC	Indian Army. The Indian Army has a regimental system, but is operationally and geographically divided into seven commands, with the basic field formation being a division. Army. Within a national military force, the word army may also mean a field army. An army (from Latin arma "arms , weapons" via Old French armée , "armed" (feminine)) or ground force is a fighting force that fights primarily on land. → The Indian Army is a military force.	Armenian Genocide. Other indigenous and Christian ethnic groups such as the Assyrians and the Ottoman Greeks were similarly targeted for extermination by the Ottoman government in the Assyrian genocide and the Greek genocide, and their treatment is considered by some historians to be part of the same genocidal policy. → The Armenian Genocide was the extermination of Armenians who were mostly Ottoman citizens.
RD	Pray for the victims. Deadly terrorist attack on French magazine Charlie Hebdo in Paris #FreePress http://t.co/HCEG92Zxtz @Parazhit @nickyromero look @Parazhit just because they published, 9 year ago, a satirical drawing of Mahomet,... One of the terrorist said "The prophet was avenged"... RT @Parazhit: Pray for the victims. Deadly terrorist attack on French magazine Charlie Hebdo in Paris #FreePress http://t.co/TrYGr2Sm1O @Parazhit Praying for Paris and France you are our brothers and sisters #EDM better days will come thanks to God and music! @Parazhit @HardRavers merci	After the attack, the gunmen shouted: "We have avenged the Prophet Mohamed! We have killed Charlie Hebdo!" http://t.co/DgmB9jTXx7 @nytimes Did they really avenge. Does the Prophet need avenging? @nytimes No cure for crazy. @nytimes Killing one Charlie has only created thousands more. #JeSuisCharlie #FreedomOfSpeech @nytimes Ironically they have given Charlie Hebdo martyr status...#JeSuisCharlie @nytimes Report this:Americans DON'T want to close Gitmo or release terrorists&WANT pipeline&borders secured.Obama not listening to ppl. @nytimes Given you're filtering victim accounts @nytimes, shocked you haven't made the killers the heroes yet. #Journalism (...)

Table 4: Examples of credible and non-credible content in each of the tasks in BODEGA: style-based news bias assessment (HN), propaganda detection (PR), fact checking (FC) and rumour detection (RD). See main text for the data sources.