SALSA: Salience-Based Switching Attack for Adversarial Perturbations in Fake News Detection Models

Chahat Raj*, Anjishnu Mukherjee*, Hemant Purohit,
Antonios Anastasopoulos, and Ziwei Zhu

George Mason University, Fairfax, Virginia, USA
{craj,amukher6,hpurohit,antonis,zzhu20}@gmu.edu
*These authors contributed equally to this work.

Abstract. Despite advances in fake news detection algorithms, recent research reveals that machine learning-based fake news detection models are still vulnerable to carefully crafted adversarial attacks. In this landscape, traditional methods, often relying on text perturbations or heuristic-based approaches, have proven insufficient, revealing a critical need for more nuanced and context-aware strategies to enhance the robustness of fake news detection. Our research identifies and addresses three critical areas: creating subtle perturbations, preserving core information while modifying sentence structure, and incorporating inherent interpretability. We propose SALSA, an adversarial **Sal**ience-based **S**witching **A**ttack strategy that harnesses salient words, using similarity-based switching to address the shortcomings of traditional adversarial attack methods. Using SALSA, we perform a two-way attack: misclassifying real news as fake and fake news as real. Due to the absence of standardized metrics to evaluate adversarial attacks in fake news detection, we further propose three new evaluation metrics to gauge the attack's success. Finally, we validate the transferability of our proposed attack strategy across attacker and victim models, demonstrating our approach's broad applicability and potency. Code and data are available here at https://github.com/iamshnoo/salsa.

Keywords: Adversarial attacks · Robustness · Interpretability · Fake news · Transformers.

1 Introduction

The proliferation of misinformation and its potential societal impact has fueled significant research in data mining and information retrieval, particularly in developing methods to detect and mitigate false information. Fake news detection presents a complex challenge due to the nuanced and evolving nature of misinformation, requiring sophisticated algorithms and an understanding of both linguistic subtleties and contextual cues. Recent breakthroughs in language models, particularly transformers [24], capable of understanding subtle contexts, have proven remarkably adept in fake news detection. But, are these systems robust? Adversarial attacks designed to cause input perturbations can depreciate their performances significantly, highlighting underlying vulnerabilities [13, 26].

Existing studies on adversarial attacks present certain limitations in their methodologies. Prior work largely depended on heuristics that involve methods such as negation [5], adverb intensity modification [5], and fact distortion [29]. These techniques tend to compromise the fundamental information of the text, creating alterations that might be significant and potentially detectable. Moreover, many of these previous strategies do not provide inherent interpretability in their attack approach, leaving a need for post-hoc explanations to understand the model's altered behavior [25]. We identify three pivotal elements central to our approach in the context of fake news detection. First, the adversarial perturbations must be crafted so that they are subtle, but still lead to significant changes in the model outputs [7]. Second, we want to preserve the text's core information, instead of making significant changes to the content of the sentence. Third, we advocate for an approach where the attack itself is inherently interpretable, negating the need for post-hoc explanations. Towards this, we propose leveraging model interpretability to precisely discern the influence of individual tokens in a news article on the model's predictions. This insight enables us to manipulate the model into producing incorrect outputs by strategically altering the identified critical tokens. In this manner, we create a subtle, content-preserving, and interpretable attack strategy. Our study makes the following contributions:

1. We conceptualize the issue of adversarial attack in fake news detection as a multifaceted problem, addressing two distinct sub-problems: the misclassification of real news articles as fake, and the misclassification of fake news articles as real. This dual-sided approach contrasts with traditional methods predominantly focusing on the misclassification of false information only.
2. We propose to leverage model interpretation as a powerful weapon to reveal the vulnerabilities of a fake news detection model and generate strong attacks. Specifically, we introduce SALSA as an implementation of our idea, which generates input perturbations by implementing similarity-based switching of words according to their salience. This approach subtly modifies the text without necessarily altering the underlying information, effectively overcoming limitations in previous methods.
3. We propose three metrics comprehensively assessing different and complementary aspects of the "success" of an attack strategy.
4. We conduct extensive experiments to evaluate our proposed method on 2 real-world datasets and observe significant improvement in attack success compared to previous methods. Besides, we also perform a study concerning the transferability of our attack strategy, further highlighting our approach's adaptability and potential applications within the landscape of fake news detection.

2 Related Work

The field of fake news detection has witnessed an exponential growth of methods aimed at identifying and mitigating false information [21, 22, 20, 27, 28, 15]. While these techniques offer promising solutions, they also expose key limitations that underline the need for further refinement and innovation [8].

There have been numerous advancements toward adversarial attacks in text classification [4, 3, 16, 23]. Methods such as DeepWordBug [6] and TextBugger [11] introduce noise into the data through input perturbations, disrupting the original context of the text and leading to unrealistic manipulations [17]. These perturbations, also observed by Ali et al. [2], present challenges in enhancing robustness, particularly as adversarial examples often rely on fewer words.

Another critical issue lies in the modification of factual information. Flores and Hao [5] highlighted vulnerabilities by attacking fake news detectors through changes in compositional semantics and lexical relations. However, their attacks, along with those by Zhou et al. [29] may distort the factual content of news articles, thus risking the integrity of the information.

Some existing methods, such as Probability Weighted Word Saliency (PWWS) [18], reduce classification accuracy through random heuristic-based word substitutions. While effective, they may alter the context or introduce artificial noise.

Moreover, most of the research until now has focused on fake-attack only, i.e., misclassifying fake news as true [10]. This one-sided view overlooks the nuanced complexity of adversarial behavior. In addition, the reliance on evaluation metrics like flip rate [5], does not necessarily capture all the elements of the attack, pointing to a need for more comprehensive attack evaluation metrics.

In response to these limitations, there is a clear need for an approach that preserves the authenticity and subtlety of the original content while effectively challenging fake news detectors. The existing methods often suffer from changing the context of the news article, introducing noise, or distorting factual information. Recognizing these challenges, we propose SALSA, a similarity-switching approach based on salience scores. Unlike previous methods, SALSA targets a subtle yet efficient attack by simply switching target tokens with candidate words using cosine similarities, replacing the words with their most similar counterparts in the embedding space. This strategy aims to maintain the content and integrity of the news article while introducing a nuanced challenge to the existing detection mechanisms. Moreover, we present a two-faceted view to our approach by modifying information both ways to perform an attack—misclassifying real news as fake and misclassifying fake news as real. Recognizing the limitations of existing metrics, we also propose three new attack evaluation metrics, designed to provide a more in-depth and accurate assessment of the attack landscape.

3 Methodology

To overcome the limitations of the previous approaches, we first define the attack in terms of our desired goals, before proposing our approach for the different components of our algorithm as illustrated in Fig. 1.

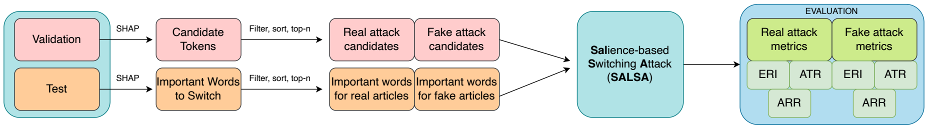

Fig. 1. The pre-requisites for SALSA involves extracting particular words of interest from the validation split to get a pool of attack candidates, and words with high salience scores for switching in the test split. These inputs are then fed into SALSA, along with the article to be perturbed to perform an attack.

3.1 Formalization and Evaluation

Given a news article a with a ground truth label y and predicted label before attack $\hat{y}$, where y and $\hat{y}$ can be either real (1) or fake (0), we have two attack goals. The objective of the attack on a real news article is to manipulate the content in such a way that the manipulated article a' is classified as fake, resulting in a new predicted label $\hat{y}' = 0$. Conversely, the objective of the attack on a fake news article is to alter the content so that the manipulated article a' is classified as real, leading to a new predicted label $\hat{y}' = 1$.

We aim to flip the labels (i.e., $\hat{y}$ should be different from $\hat{y}'$ after an attack procedure T is performed), but specifically for those labels that contribute towards misclassification. We can define this formally as two cases using the example of a "real attack" scenario:

Case 1: The original classification is correct (i.e., $y = 1, \hat{y} = 1$), we want to flip the labels such that: $T(a, y = 1, \hat{y} = 1) \rightarrow a', \hat{y}' = 0$

Case 2: The original classification is incorrect (i.e., $y = 1, \hat{y} = 0$), we do not seek to flip the labels, and thus: $T(a, y = 1, \hat{y} = 0) \rightarrow a', \hat{y}' = 0$

Here, a' is the modified article after procedure T, and $\hat{y}'$ is the final predicted label. T is designed to selectively flip the labels of the first type of articles that were originally getting correctly classified, leading to a misclassification.

We propose three metrics designed to measure the effectiveness of adversarial attacks in the context of fake news classification: Error Rate Increase (ERI), Attack Turning Rate (ATR), and Attack Reserving Rate (ARR). These are percentage values with the maximum being 100, indicating an ideal attack. For all three metrics, the higher the value of the metric, the more the effectiveness of the attack method.

Error Rate Increase: ERI quantifies the percentage growth in the number of errors committed by the model as a result of the input perturbations introduced through the attack.

$$ERI_{\text{real}} = Error_{\text{real}_{(\text{after})}} - Error_{\text{real}_{(\text{before})}}, \qquad Error_{\text{real}} \quad = FN/(FN + TP) \quad (1)$$

$$ERI_{\text{fake}} = Error_{\text{fake}_{(\text{after})}} - Error_{\text{fake}_{(\text{before})}}, \qquad Error_{\text{fake}} \quad = FP/(FP + TN) \quad (2)$$

where FN, FP, FN, TP denote the counts of false negatives, false positives, true positives, and true negatives, respectively.

Attack Transition Rate: ATR quantifies the count of news items that undergo a "transition" from being correctly classified prior to the attack to being misclassified subsequently.

$$ATR_{real} = \frac{|y = 1, \hat{y} = 1, \hat{y}' = 0|}{|y = 1, \hat{y} = 1|}, \qquad ATR_{fake} = \frac{|y = 0, \hat{y} = 0, \hat{y}' = 1|}{|y = 0, \hat{y} = 0|} \tag{3}$$

Attack Reserving Rate: ARR quantifies the number of news items that remain misclassified after the attack, where the input perturbations do not reverse the predicted labels, but instead "reserve" them in their incorrect state.

$$ARR_{real} = \frac{|y = 1, \hat{y} = 0, \hat{y}' = 0|}{|y = 1, \hat{y} = 0|}, \qquad ARR_{fake} = \frac{|y = 0, \hat{y} = 1, \hat{y}' = 1|}{|y = 0, \hat{y} = 1|} \tag{4}$$

Motivation To perturb the article a to obtain a' in alignment with the required goals of attack, we emphasize the strategic use of model interpretability. Our approach identifies the model's vulnerabilities, which can then be exploited to manipulate the output. In line with this approach, we present SALSA as an implementation to demonstrate our interpretability-based attack strategy. This method functions by substituting important words w, which contribute to the model's confidence towards the predicted label $\hat{y}$, within the article a, with words w' from a collection of candidate attack tokens. The operation is defined by:

$$\mathbf{SALSA}(a, w', w) \rightarrow a' \tag{5}$$

Section 3.2 describes the process by which we determine the candidate attack tokens w', section 3.3 details our process of choosing important words w and finally section 3.4 gives an outline of the algorithm for replacing w with w' based on semantic similarity.

3.2 Candidate Token Generation

The process of adversarial attack on the fake news detection model entails generating suitable candidate tokens to deceive the model into misclassifying real news as fake (real attack) and fake news as real (fake attack).

Table 1. Given a predicted label, we selectively choose attack candidates based on their salience scores for each type of attack.

	Real Attack Candidates	**Fake Attack Candidates**
Predicted Label: $(\hat{y} = 0)$	+ Salience Scored Tokens	- Salience Scored Tokens
Predicted Label: $(\hat{y} = 1)$	- Salience Scored Tokens	+ Salience Scored Tokens

Salience score assignment using SHAP To find words of importance in the news articles in the validation split, we use interpretability as a tool to drive our token generation procedure. Primarily, we employ SHAP [12] to find words with high salience scores for every news article. To do so, SHAP uses the model m trained on the training split, and also the labels L predicted by that model for the items in the validation split. Formally,

$$SHAP(a, m, \hat{y}) \rightarrow [+, -] \quad \forall \hat{y} \in L \tag{6}$$

Significance of salience scores For a given model prediction $\hat{y}$, a positive salience score for a word indicates that the presence of the word in the article increases the confidence of the model towards that specific prediction. If these words were to be removed from the article, the model's confidence for the prediction being $\hat{y}$ would reduce significantly, and at some point, the model would find more confidence in predicting the flipped label for the same article.

Global candidate pool We aim to find tokens contributing most to the model's decision for both fake and real news articles so that we can later use them to guide the decision making process towards incorrect predictions on articles from the test split. We will create a candidate set for real attack and another set for fake attack. To do so, we use Equation 6 to generate a pool of all words with positive salience scores and all words with negative salience scores in the validation split. There are however many articles in this split, resulting in a very large initial pool size. So, we develop a strategy to locate the most important words overall, across all articles. Considering the case of generating real attack candidate tokens, we utilize words with positive salience scores from news items that were predicted as fake ($\hat{y} = 0$) and words with negative salience scores from news items that were predicted as real ($\hat{y} = 1$), both indicate words instrumental for the sentence being classified as fake. Table 1 summarizes the type of salience-scored words used for real and fake attack candidate generation. Only a few tokens per sentence contribute towards the model's decision making process, while the bulk of the remaining tokens make up a very small combined effect. To reflect this on a global scale, across all the articles, we sort the candidate pool based on the absolute values of the salience scores and choose the top N, where N is treated as a hyperparameter that we study extensively in Table 3.

3.3 Target Token Selection

The next step of SALSA is the selection of important target tokens for switching. These are words that contribute significantly to the confidence of the model in making its predictions.

First, we use SHAP with our trained model m on articles in the test split to get corresponding positive and negative salience scores for each word in an article similar to candidate token generation.

However the strategy of choosing important words from the initial pool is different. Here, we select a specific number, M, of words for each article. We study different possible values of the hyperparameter M, in conjunction with the hyperparameter N for candidate token generation in Table 3. The following outlines the approach for each scenario based on our goal for the attack:

1. $\mathbf{y = 0} \mid \hat{\mathbf{y}} = \mathbf{0}$ | **Fake Attack:** The prediction needs to change from fake to real. Hence, +ve salience words supporting fake prediction are switched.
2. $\mathbf{y = 0} \mid \hat{\mathbf{y}} = \mathbf{1}$ | **Fake Attack:** The prediction should remain as real. Negative salience-scored words opposing the real prediction are removed.
3. $\mathbf{y = 1} \mid \hat{\mathbf{y}} = \mathbf{0}$ | **Real Attack:** The prediction should remain as fake. Negative salience words and words with the smallest absolute +ve scores are removed.
4. $\mathbf{y = 1} \mid \hat{\mathbf{y}} = \mathbf{1}$ | **Real Attack:** The prediction needs to change from real to fake. Hence, +ve salience words that support the real prediction are switched.

3.4 SALSA

Given the set of candidate tokens, and the target tokens for each article in the test set, we can now formulate our algorithm for input perturbations.

The SALSA algorithm is designed to semantically alter a given article by selectively substituting significant tokens with closely related alternatives from a set of candidates, utilizing word embeddings and cosine similarity for guidance. It starts by transforming words in the target set w and candidate tokens in w' into embeddings (E_w and $E_{w'}$) using a transformer model, capturing their semantic essence in a high-dimensional space. A similarity matrix κ is then derived from the cosine similarity between $E_{w'}$ and E_w. Subsequently, for each vital word w_i in w, tokens in w' are ranked based on their similarity to w_i, excluding w_i itself to avoid redundancy, and stored in sorted order. The algorithm proceeds to replace w_i with the highest-ranked, distinct candidate token $\bar{w}$ in the article, iterating this process for all words in w to generate a semantically similar, yet modified version of the original article.

Algorithm 1 Switch Words Using Salience-based Switching Attack (SALSA)

```
procedure SALSA(a, $w'$, w)    ▷ a: Article, $w'$: Candidate Tokens, w: Important Words
    $E_w \leftarrow$ Embeddings of $w$
    $E_{w'} \leftarrow$ Embeddings of $w'$
    $\kappa \leftarrow$ Cosine Similarity between $E_{w'}$ and $E_w$
    for each $w_i$ in $w$ do
        $z \leftarrow$ Sorted indices of words similar to $w_i$ in $\kappa$
        $\bar{w} \leftarrow$ The highest ranked $w'_k \in w'$ in $z$ where $w'_k \neq w_i$
        Replace $w_i$ with $\bar{w}$ in $a$
    end for
    return $a'$
end procedure
```

4 Experiments

We benchmark our approach on two datasets: NELA-GT-2018 [14] and TFG [1]. A lightweight DistilBERT [19] model fine-tuned for four epochs, achieves 91.65%

and 99.04% classification accuracies on the test splits for both the datasets. For each dataset, we then implement state-of-the-art fake news detection attack baselines and compare them with our proposed algorithm SALSA for both real and fake attacks. We report the results using our three novel evaluation metrics ERI, ATR, and ARR (higher values indicating more effective attack). Further, a hyperparameter study is presented by varying the number of target tokens for switching and the number of candidate tokens for attack. Additionally, we conduct an attack-transferability study by interchangeably using DistilBERT and RoBERTa for different stages of our algorithm. Our goal is to investigate the following research questions:

- **RQ1:** How does SALSA perform compared to state-of-the-art baselines?
- **RQ2:** How do hyper-parameters influence the attack performance?
- **RQ3:** How effective is SALSA in transfer of attacks across different classification models?

Table 2. SALSA outperforms baselines for both real and fake attack on the NELA-GT-2018 dataset. For TFG, it is better for real attack, but for fake attack, the performance is comparable to the negation attack and adverb intensity baselines.

	NELA-GT-2018 Dataset						TFG Dataset					
Attack Strategy	Real Attack			Fake Attack			Real Attack			Fake Attack		
	ERI	ATR	ARR	ERI	ATR	ARR	ERI	ATR	ARR	ERI	ATR	ARR
Negation	24.97	31.14	67.98	16.18	20.75	56.96	11.69	11.85	91.43	2.24	3.13	25.58
Adverb Intensity	25.13	31.30	68.08	16.47	21.04	57.28	11.39	11.55	91.43	2.34	3.21	27.91
Random Injection	33.55	39.82	75.71	17.25	21.97	56.10	16.13	16.38	85.71	8.50	9.35	34.88
Head Injection	35.98	42.68	74.11	24.82	29.66	**61.99**	**72.39**	**73.02**	**94.29**	**24.35**	**25.26**	**46.51**
Tail Injection	34.36	40.68	76.08	19.72	24.72	54.93	13.43	13.58	94.29	6.13	6.93	37.21
Similarity Switching	34.48	41.03	74.01	20.30	25.44	53.64	14.44	14.62	91.43	11.83	12.64	41.86
SALSA	**51.01**	**57.92**	**87.19**	**26.17**	**31.18**	61.24	67.55	68.21	85.71	7.51	8.36	34.88

4.1 Experimental setup

Datasets. The NELA-GT-2018 dataset, [14], short for "News Landscape Ground Truth 2018", is a rich and expansive resource for news content analysis. Comprised of around 713,000 articles collected from 194 news sources, it covers an entire year of news reporting, from February 2018 to February 2019. The dataset is suitable for a range of tasks in media bias analysis, misinformation detection, and content understanding.

The TFG dataset [1] provides a valuable resource for research on automatic fake news detection. The dataset contains approximately 40k text news articles, evenly divided into reliable and unreliable/fake classes. The data covers a wide range of news topics, sourced from known publishers of mainstream news as well as propaganda and fake news sites.

We use both of these to frame fake news detection as a classification problem, training a classifier to predict whether the input text is "real" or "fake", using a 70:15:15 train-validation-test split.

Baselines. We explore two existing baselines for this challenge, as introduced in the paper by Flores and Hao [5]. The first approach of **negation attack**, identifies third-person singular subjects and negates the associated first-person verbs. The second method, **adverb intensity attack**, eliminates the polarizing adverbs from the articles. Given that both NELA and TFG often feature lengthy articles, these straightforward alterations to the semantic content of the article generally do not significantly influence the model's predictions.

Further, we try two different heuristic approaches that can be considered simplified versions of our final algorithm SALSA. The first approach of **injection attacks** only creates a candidate set similar to SALSA, but instead of switching any words in the article, it simply injects a fixed number of words (10) into the article. The injection location can be chosen randomly, or all of the words can be injected at either the head of the article or the tail, leading to the three variations of the injection attack. The head injection attack can be considered a control experiment, because transformer models are especially effective at capturing context at the start of long articles, and we are intentionally modifying that initial context with multiple attack candidates. The second naive approach is to perform **similarity switching** with a candidate set similar to SALSA, but without specifically identifying important words to switch in the article.

Finally, we incorporate both the candidate set generation strategy and the important token selection strategy into our SALSA algorithm which switches important tokens with similar ones from the candidate set.

Reproducibility. We fine-tuned two pretrained models available on huggingface, "distilbert-base-uncased"[1] and "roberta-base"[2]. For both, we added a classification head projecting the 768 dimensional output onto a 64 dimensional space, and using dropout probability of 0.4 before predicting the probabilities of the classification labels. We perform the fine-tuning process for 4 epochs and a batch size of 64. For tokenization purposes, we limit maximum length of a sequence to 512 as most articles fit within that size. We use Adam optimizer [9] with a learning rate of 5e-5 and an epsilon of 1e-8. All experiments are executed with a fixed random seed of 42. For the results of SALSA shown in Table 2, we have fixed the number of words to switch to 20 and the size of the candidate pool as 100. The values corresponding to this setting is highlighted in Table 3. Also, for all the attacks shown in Table 2, the candidate pool uses corresponds to the "Mixed POS" setting from Fig. 2, where no POS filtering is done. Code and results are available at https://github.com/iamshnoo/salsa.

4.2 Performance of SALSA (RQ1)

A consistent observation across all conducted experiments is the pronounced difference in the difficulty of executing fake attacks as compared to real attacks. This discrepancy is delineated in Table 2, where the ERI for real attacks invariably registers a higher percentage than that for fake attacks. A plausible explanation for this phenomenon may be grounded in the distinct semantic characteristics that differentiate fake news articles from genuine ones. Real news

[1] https://huggingface.co/distilbert-base-uncased

[2] https://huggingface.co/roberta-base

articles frequently contain key indicators, such as references to reputable sources like “Reuters”, “Fox”, “CNN”, and they are often composed in a stylistically recognizable manner. Modern transformer architectures, trained on extensive data sets, are adept at discerning these subtleties, as is observed from high values of accuracy and other classification metrics on both the datasets that we analyzed. Consequently, deceiving the model into misclassifying a fake article as real presents a substantial challenge, depending on the selected attack method. Conversely, inducing a model to falsely label a real article as fake proves to be a relatively easier task.

An additional significant finding evident from Table 2 pertains to the contrasting performance of SALSA across two datasets: NELA-GT-2018 and TFG. In the case of NELA-GT-2018, SALSA outperforms other methods for both fake and real attacks, establishing itself as the most effective strategy. Conversely, for the TFG dataset, SALSA’s performance in real attacks is comparable to the control setting of head injection, whereas its efficacy in fake attacks does not present a substantial improvement over other approaches. This can be attributed to the fact that the average article length for TFG is shorter than that for NELA-GT-2018, and consequently fake news articles are much easier to detect due to their semantic structure. Moreover, the classifier trained for TFG exhibits a remarkable 99% accuracy on the test set, in contrast to 91.6% for the NELA-GT-2018. Such a high degree of accuracy for TFG implies a robust generalization by the model, rendering it resilient to deception, particularly when restricted to implementing only subtle input perturbations.

4.3 Hyperparameter study (RQ2)

To study the effect of varying the hyperparameters N and M, corresponding to the number of candidate tokens in the attack set, and the number of important tokens per article that we want to switch respectively, we perform an extensive evaluation in Table 3. We find that, choosing a large number of words (more than 30) for switching usually improves ERI for the same candidate set. However, we still limit to about 20 words to remain within our constraints of performing “subtle” perturbations. For the candidate set, larger is not necessarily better. In fact, we see many cases where having a set of 50 or 100 candidates performs better than having 200 candidates. This can be owed to the fact that the tokens in the candidate set were chosen after having been sorted by absolute salience score, so for a larger set we might end up having some less meaningful words in the set that do not affect the predictions as much as the top 50 would for instance. The final hyperparameters that we chose for our experiments were $M = 20, N = 100$, which gives subtle enough perturbations while being as optimal as possible in terms of performance.

4.4 Effect of using different POS for attack tokens (RQ2)

We perform another study that considers the effects of having different parts of speech in the candidate set. Figure 2 shows the results for real attack and fake

Table 3. Varying the number of words to switch and the size of the candidate pool shows that for the same candidate set size, switching more words leads to better attacks. Also, a smaller candidate pool with more relevant words performs better than larger sets, indicating the importance of choosing attack tokens carefully.

		NELA-GT-2018 Dataset						TFG Dataset					
		Real Attack			Fake Attack			Real Attack			Fake Attack		
(M)	(N)	ERI	ATR	ARR	ERI	ATR	ARR	ERI	ATR	ARR	ERI	ATR	ARR
5	25	33.17	39.37	75.89	20.38	25.27	57.07	55.06	55.58	91.43	4.08	4.96	27.91
5	50	32.19	38.41	74.76	18.16	22.95	56.10	55.06	55.60	88.57	5.30	6.17	30.23
5	100	33.13	39.34	75.80	23.17	28.00	60.49	47.53	48.02	85.71	4.88	5.71	32.56
5	150	31.87	38.13	74.11	22.60	27.35	61.03	34.05	34.44	85.71	5.28	6.12	32.56
5	200	31.73	38.00	73.82	23.17	27.97	60.92	34.19	34.58	85.71	5.09	5.96	30.23
10	25	39.37	45.83	79.66	19.45	24.37	55.67	66.38	67.04	85.71	4.29	5.18	27.91
10	50	38.16	44.64	78.34	18.33	23.24	54.60	67.30	67.91	91.43	7.27	8.11	34.88
10	100	39.48	46.01	79.19	24.79	29.61	62.31	55.77	56.36	82.86	5.94	6.77	34.88
10	150	37.55	44.04	77.59	23.50	28.43	59.53	38.02	38.48	80.00	7.30	8.14	34.88
10	200	37.02	43.63	75.99	24.68	29.45	62.85	38.13	38.55	85.71	6.95	7.79	34.88
20	25	49.31	56.15	86.06	15.90	21.30	45.93	78.37	79.14	82.86	5.06	5.98	25.58
20	50	47.35	54.19	84.27	15.07	20.63	42.93	82.01	82.77	88.57	9.78	10.59	39.53
20*	**100***	**51.01**	**57.92**	**87.19**	**26.17**	**31.18**	**61.24**	**67.55**	**68.21**	**85.71**	**7.51**	**8.36**	**34.88**
20	150	46.94	53.75	84.09	24.81	29.96	58.03	44.41	44.88	85.71	9.51	10.38	34.88
20	200	46.44	53.36	82.58	27.98	33.03	62.53	43.86	44.31	88.57	9.78	10.67	32.56
30	25	56.74	63.91	90.49	12.29	18.27	34.69	83.00	83.78	85.71	5.68	6.63	23.26
30	50	54.51	61.76	87.57	12.18	18.16	34.58	88.57	89.35	91.43	12.82	13.72	34.88
30	100	59.16	66.52	91.15	26.20	31.55	56.85	73.58	74.29	85.71	9.70	10.57	34.88
30	150	54.08	61.30	87.38	25.26	30.74	54.28	50.16	50.68	85.71	12.98	13.88	34.88
30	200	53.34	60.60	86.35	28.77	34.07	60.06	49.52	50.01	88.57	14.02	14.93	34.88
40	25	63.51	71.15	92.94	8.39	14.98	23.02	85.01	85.82	85.71	6.77	7.79	18.60
40	50	61.03	68.66	90.58	8.90	15.51	23.23	90.81	91.61	91.43	17.11	18.11	30.23
40	100	66.21	74.02	94.07	24.39	30.21	49.04	76.37	77.11	85.71	14.10	14.99	37.21
40	150	61.04	68.64	90.77	24.38	30.36	46.90	57.20	57.77	85.71	18.47	19.35	41.86
40	200	60.44	68.17	89.08	28.72	34.56	53.00	55.84	56.39	88.57	21.50	22.43	41.86

Table 4. Using different models for generating the inputs to SALSA via SHAP, and predicting the labels after attack, we find the larger model (RoBERTa) to be more susceptible to attacks generated using the smaller model.

	DistilBERT-attack						RoBERTa-attack					
	Real Attack			Fake Attack			Real Attack			Fake Attack		
	ERI	ATR	ARR	ERI	ATR	ARR	ERI	ATR	ARR	ERI	ATR	ARR
DistilBERT-victim	51.01	57.92	87.19	26.17	31.18	61.24	*33.32*	*40.03*	*71.37*	*11.15*	*15.73*	*51.82*
RoBERTa-victim	*71.66*	*78.44*	*95.62*	*6.04*	*12.96*	*11.46*	16.25	18.55	90.45	3.60	5.32	80.11

attack for NELA-GT-2018. The default setting that we explained in the results previously and also for our hyperparameter study contains candidate tokens with all types of parts of speech mixed in. But if we filter this set and only include candidate tokens that are verbs, we observe a higher attack ERI for real articles from NELA-GT-2018. But for fake attack, using only verbs does not perform as well as using only nouns or using only adverbs and adjectives for example. This leads us to conclude that different parts of speech might be more relevant for real attack and fake attack, and this setting might also vary across datasets.

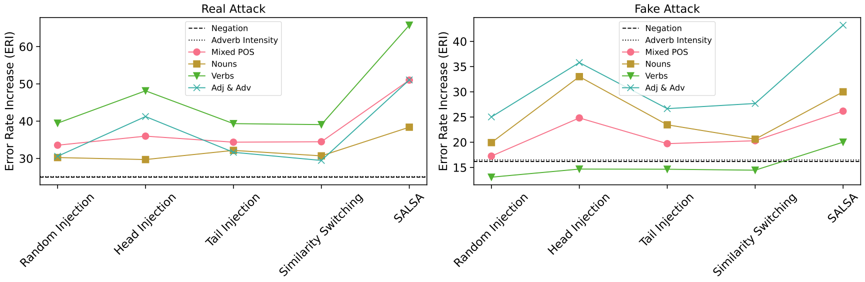

Fig. 2. Keeping only one part of speech as attack candidates has varying performances across different types of attacks.

4.5 Attack Transferability (RQ3)

While our results clearly show that SALSA works well under different hyperparameter settings for both fake and real attacks across datasets, the model m_1 that we used for generating SHAP outputs for candidate tokens and important words to switch, was the same as the model m_2 which we used for predictions before and after the attack, i.e $m_1 = m_2$.

For attack transferability, we consider 2 distinct models m_1 (DistilBERT) and m_2 (RoBERTa), such that one of the models has much fewer parameters than the other. We would be using one of the models ("attacker model") to generate the inputs required by SALSA and then use the other model ("victim model") to predict labels on the perturbed inputs. Comparing this result with the basic setting of using the same model for both tasks, we get Table 4.

When we use the smaller model m_1 for predictions after generating SALSA inputs using the bigger model m_2, it is observed that all the three attack metrics fall for both real and fake attack compared to the default case of using m_1 for both predictions and SALSA. However, when we use the bigger model m_2 for predictions after using m_1 for SALSA, we notice an increase in the metrics, with a very notable increase in the ERI for real attack. Based on these experiments, our understanding is that bigger models are more susceptible to attacks generated using SALSA and a smaller model, but the reverse is not necessarily true.

5 Discussion

The proposed adversarial attack strategy represents an important step in identifying attacks and defending against them. Since our approach is interpretable, it would enable an easier understanding of adversarial inputs, which cause the model to flip predictions. Also, it would encourage the development of methods of defending against such subtle and interpretable attacks. Further, our results on attack transferability, which shows the susceptibility of larger models for words generated from smaller ones, can be a starting point to audit the vulnerabilities of large language models that have increasing usage over time.

6 Conclusion and Future Work

In this research, we conducted a thorough examination of adversarial attack strategies on fake news detection, with a specific focus on the inherently interpretable SALSA algorithm. Our findings underscored the nuanced challenges in executing fake attacks compared to real ones and highlighted the contrasting performance of SALSA across different datasets and semantic structures. In subsequent research, we aim to broaden the comparative framework by incorporating alternative interpretability methods such as LIME and Integrated Gradients.

A Appendix

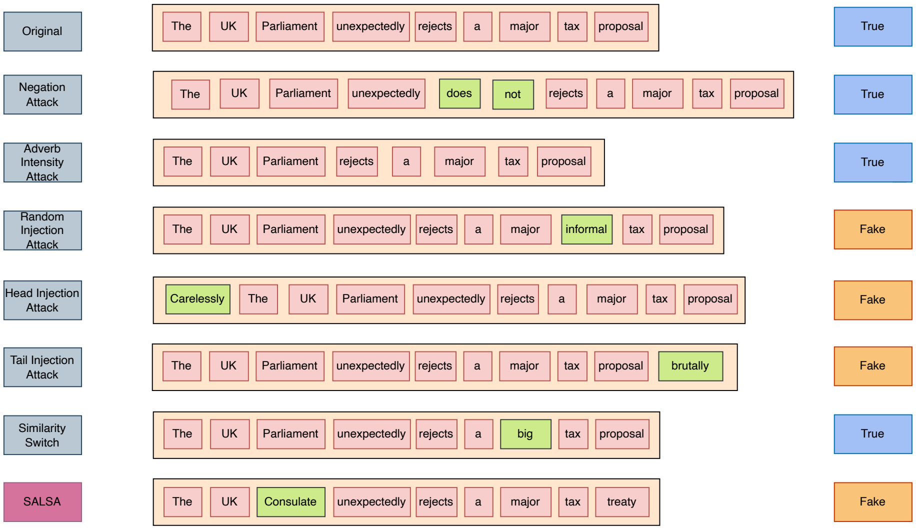
Original
The UK Parliament unexpectedly rejects a major tax proposal
True
Negation Attack
The UK Parliament unexpectedly does not rejects a major tax proposal
True
Adverb Intensity Attack
The UK Parliament rejects a major tax proposal
True
Random Injection Attack
The UK Parliament unexpectedly rejects a major informal tax proposal
Fake
Head Injection Attack
Carelessly The UK Parliament unexpectedly rejects a major tax proposal
Fake
Tail Injection Attack
The UK Parliament unexpectedly rejects a major tax proposal brutally
Fake
Similarity Switch
The UK Parliament unexpectedly rejects a big tax proposal
True
SALSA
The UK Consulate unexpectedly rejects a major tax treaty
Fake

Fig. 3. Comparison of attack strategies on an example article with $y = 1$ (real attack)

Bibliography

[1] https://huggingface.co/datasets/GonzaloA/fake_news

[2] Ali, H., Khan, M.S., AlGhadhban, A., Alazmi, M., Alzamil, A., Al-Utaibi, K., Qadir, J.: All your fake detector are belong to us: evaluating adversarial robustness of fake-news detectors under black-box settings. IEEE Access **9**, 81678–81692 (2021)

[3] Chang, G., Gao, H., Yao, Z., Xiong, H.: Textguise: Adaptive adversarial example attacks on text classification model. Neurocomputing **529**, 190–203 (2023)

[4] Ebrahimi, J., Rao, A., Lowd, D., Dou, D.: Hotflip: White-box adversarial examples for text classification (2017), arXiv:1712.06751

[5] Flores, L.J.Y., Hao, Y.: An adversarial benchmark for fake news detection models (2022), arXiv:2201.00912

[6] Gao, J., Lanchantin, J., Soffa, M.L., Qi, Y.: Black-box generation of adversarial text sequences to evade deep learning classifiers. In: 2018 IEEE Security and Privacy Workshops (SPW). pp. 50–56. IEEE (2018)

[7] Ghaffari Laleh, N., Truhn, D., Veldhuizen, G.P., Han, T., van Treeck, M., Buelow, R.D., Langer, R., Dislich, B., Boor, P., Schulz, V., et al.: Adversarial attacks and adversarial robustness in computational pathology. Nature communications **13**(1), 5711 (2022)

[8] Horne, B.D., Nørregaard, J., Adali, S.: Robust fake news detection over time and attack. ACM Transactions on Intelligent Systems and Technology (TIST) **11**(1), 1–23 (2019)

[9] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2014), arXiv:1412.6980

[10] Koenders, C., Filla, J., Schneider, N., Woloszyn, V.: How vulnerable are automatic fake news detection methods to adversarial attacks? (2021), arXiv:2107.07970

[11] Li, J., Ji, S., Du, T., Li, B., Wang, T.: Textbugger: Generating adversarial text against real-world applications (2018), arXiv:1812.05271

[12] Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. Advances in neural information processing systems **30** (2017)

[13] Morris, J.X., Lifland, E., Yoo, J.Y., Grigsby, J., Jin, D., Qi, Y.: Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp (2020), arXiv:2005.05909

[14] Nørregaard, J., Horne, B.D., Adalı, S.: Nela-gt-2018: A large multi-labelled news dataset for the study of misinformation in news articles. In: Proceedings of the international AAAI conference on web and social media. vol. 13, pp. 630–638 (2019)

[15] Oshikawa, R., Qian, J., Wang, W.Y.: A survey on natural language processing for fake news detection (2018), arXiv:1811.00770

[16] Pan, L., Hang, C.W., Sil, A., Potdar, S.: Improved text classification via contrastive adversarial training. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 11130–11138 (2022)

[17] Pruthi, D., Dhingra, B., Lipton, Z.C.: Combating adversarial misspellings with robust word recognition (2019), arXiv:1905.11268

[18] Ren, S., Deng, Y., He, K., Che, W.: Generating natural language adversarial examples through probability weighted word saliency. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 1085–1097. Association for Computational Linguistics, Florence, Italy (Jul 2019)

[19] Sanh, V., Debut, L., Chaumond, J., Wolf, T.: Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter. CoRR **abs/1910.01108** (2019)

[20] Shu, K., Cui, L., Wang, S., Lee, D., Liu, H.: defend: Explainable fake news detection. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 395–405 (2019)

[21] Shu, K., Sliva, A., Wang, S., Tang, J., Liu, H.: Fake news detection on social media: A data mining perspective. ACM SIGKDD explorations newsletter **19**(1), 22–36 (2017)

[22] Shu, K., Wang, S., Liu, H.: Beyond news contents: The role of social context for fake news detection. In: Proceedings of the twelfth ACM international conference on web search and data mining. pp. 312–320 (2019)

[23] Simoncini, W., Spanakis, G.: Seqattack: On adversarial attacks for named entity recognition. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 308–318 (2021)

[24] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., Rush, A.: Transformers: State-of-the-art natural language processing pp. 38–45 (Oct 2020)

[25] Xu, K., Liu, S., Zhao, P., Chen, P.Y., Zhang, H., Fan, Q., Erdogmus, D., Wang, Y., Lin, X.: Structured adversarial attack: Towards general implementation and better interpretability (2018), arXiv:1808.01664

[26] Zeng, G., Qi, F., Zhou, Q., Zhang, T., Ma, Z., Hou, B., Zang, Y., Liu, Z., Sun, M.: Openattack: An open-source textual adversarial attack toolkit (2020), arXiv:2009.09191

[27] Zhang, X., Ghorbani, A.A.: An overview of online fake news: Characterization, detection, and discussion. Information Processing & Management **57**(2), 102025 (2020)

[28] Zhou, X., Zafarani, R.: A survey of fake news: Fundamental theories, detection methods, and opportunities. ACM Computing Surveys (CSUR) **53**(5), 1–40 (2020)

[29] Zhou, Z., Guan, H., Bhat, M.M., Hsu, J.: Fake news detection via nlp is vulnerable to adversarial attacks (2019), arXiv:1901.09657