2023 2nd International Conference on Vision Towards Emerging Trends in Communication and Networking Technologies (ViTECoN) | 979-8-3503-4798-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/VITECON58111.2023.10157098

Bypassing Deep Learning based Sentiment Analysis from Business Reviews

Ashish Bajaj
Biometric Research Laboratory, Department of Information Technology, Delhi Technological University
Bawana Road, Delhi-110042, India
bajaj.ashish25@gmail.com

Dinesh Kumar Vishwakarma
Biometric Research Laboratory, Department of Information Technology, Delhi Technological University
Bawana Road, Delhi-110042, India
dvishwakarma@gmail.com

Abstract— **In recent years, online reviews of businesses have grown increasingly significant, as customers and even competitors use them to evaluate a company's quality. Yelp is one of the most popular review websites, and it would be advantageous for them to be capable of predicting the sentiment or even the star rating of a review. Current deep-learning algorithms excel at sentiment classification. With the tremendous performance of models based on deep learning in text-related problems, they are susceptible to adversarial manipulations that result in inaccurate sentiment classification. An adversarial text is created by manipulating just few letters or words in such a manner so that general meaning of the text remains unchanged for humans but fooling a system into making false predictions. This study highlights the shortcomings of sentiment categorization by employing a range of cutting-edge attack techniques to generate perturbed text. We examined the performance of several models, including BERT, an advanced transformer model, and the extensively used LSTM and Word-CNN classifiers trained on the Yelp polarity dataset. For each model, Attack Success Rates (ASR) are calculated as the evaluation metric. Based on the experimental results, we determined which sentiment classifier is more vulnerable to adversarial perturbations and which is more resistant. The results demonstrate that automatic sentiment classification techniques can be circumvented, which has implications for present policy approaches.**

Keywords— (Natural Language Processing) NLP, Sentiment Classification, Adversarial Attack, Transformers, Semantic Similarity, Vulnerability.

I. Introduction

Across the past decade, Machine Learning (ML) approaches have flourished at a range of tasks, including regression, classification, and decision processing. Yet, these models are fragile to adversarial situations, which are genuine inputs that have been intentionally modified by minute, frequently imperceptible variations. Emerging research has generated adversarial perturbed images which render algorithms for computer vision ineffective[1]. A few research on adversarial cases in text-categorization problems have been undertaken, such as emotional analysis, topic classification[2], machine translation, fake news classification, hate content detection, etc. Yet, due to the adversarial machine learning's achievement in visuals, it is a relatively recent topic that has received more attention and is interesting to examine [3]. To produce adversarial cases, two hostile situations are used. A *"black-box setup"* is the design of adversarial perturbed sample in which an attacker is clueless of classification algorithm or the training information. In contrast, in a *"white-box"* situation, the latter one has detailed understanding of the algorithm and the training set [4]. In addition, the attack is also classified according to whether it is targeted or untargeted. Consider the input class to be $\boldsymbol{C_i}$ and the output class to be $\boldsymbol{C_T}$. The input $\boldsymbol{x}$ belongs to the class $\boldsymbol{C_i}$, and we desire that, after perturbation, $\boldsymbol{x'}$ set belongs to a class other than $\boldsymbol{C_i}$. In a *"targeted"* assault, the input data $\boldsymbol{x}$ is altered to $\boldsymbol{x'}$ so that it predicts a targeted class $\boldsymbol{C_T}$ rather than its genuine class $\boldsymbol{C_i}$. Here, the objective is to reach a specific target label. In an *"untargeted"* attack, the objective is to shift input $\boldsymbol{x}$ away from its true class $\boldsymbol{C_i}$, regardless of which other classes are struck [5]. Technically, in virtue to fooling the target models, the outcomes of a natural language parsing framework must fulfill 3 key utility-preserving functionalities :***1.)*** same resemblance in meaning—the produced example should have the same significance as the actual based on human judgement; ***2.)*** created instances of opposition shall seem grammatical and genuine. ***3.)*** Human predictions should be consistent and remain constant[6]. **Figure 1** demonstrates how assaults are classified depending on attack specificity and attacker knowledge.

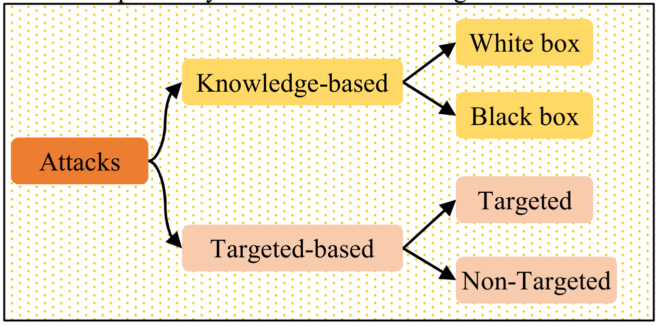
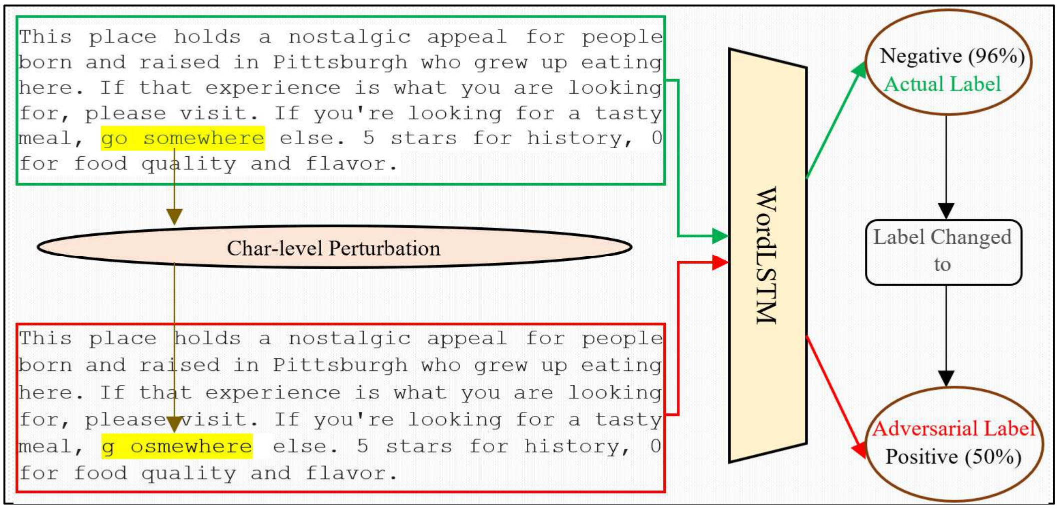


Figure 1. Taxonomy of Adversarial Attack.

This research focuses on the widely used word convolution neural-networks and bi-directional long short-term memory, along with the potent transformer model, that is, BERT, for various natural language processing tasks, to illustrate the weakness in sentiment classification. Firstly, the classifiers are trained on the (Yelp polarity dataset), a well-known set of business review emotions. The deterioration of these pre-trained models' performance is then examined by conducting attacks utilizing various cutting-edge adversarial attack methodologies. The findings may be of interest to users who habitually use well-known cutting-edge classification methods. The reader will be able to choose which model is

979-8-3503-4798-2/23/$31.00 ©2023 IEEE



This place holds a nostalgic appeal for people born and raised in Pittsburgh who grew up eating here. If that experience is what you are looking for, please visit. If you're looking for a tasty meal, go somewhere else. 5 stars for history, 0 for food quality and flavor.

This place holds a nostalgic appeal for people born and raised in Pittsburgh who grew up eating here. If that experience is what you are looking for, please visit. If you're looking for a tasty meal, g osmewhere else. 5 stars for history, 0 for food quality and flavor.

Figure 2. Adversarial instance created by modifying a character[12] in such a way that it is semantically equivalent to a human observer yet evade the Word CNN model into producing the incorrect predicted label.

most suited to their situation. Furthermore, this pushes researchers to develop models that use adversarial resilient generalizations rather than traditional generalizations. According to the authors, this is the first study to compare the sensitivity of various models to different adversarial attack approaches for the sentiment categorization task.

II. Related Work

A. *Sentiment Classification*

Sentiment classification, also referred to as information extraction, is the process of recognizing & analyzing information depending on the context in which it is expressed. This content may comprise tweets, comments, criticisms, and even passionate rants with varied or neutral perspectives. Examples of popular applications for sentiment analysis include monitoring client comments, identifying specific consumers to improve service, and analyzing how a change in a product or service affects how customers feel. Tracking customer mood over time is also beneficial. This platform has significantly transformed the way businesses operate, from opinion polls to innovative marketing strategies. Several internet recommendation algorithms, for instance, analyze user reviews and comments based on their emotional content. Classical machine learning techniques, such as Nave Bayes, Decision tree, etc., have found tremendous success in predicting sentiment from business reviews. With emerging deep learning approaches, however, models such as Bi-directional Long Short-Term Memory, Word Convolutional Neural Network and transformer models are more sophisticated and accurate[7]. As a result of their high precision and accuracy scores in text processing tasks, these models have attracted substantial interest. In this research, the most effective classification models based on deep learning were utilized. Using realistic attacks, this essay highlights the significance of testing deep learning-based sentiment classifiers prior to incorporating them into decision support systems.

B. *Adversarial Attacks in Textual-domain*

The significance of adversarial pattern recognition in modern artificial intelligence has grown in recent days. The adversarial component of adversarial machine learning is adversarial attacks. In an adversarial algorithm, the input data of a neural-network are modified to test its capacity to produce the same outcomes [8]. To provide a more intuitive comprehension of the definitions, we describe DNN and adversarial examples using formulas.

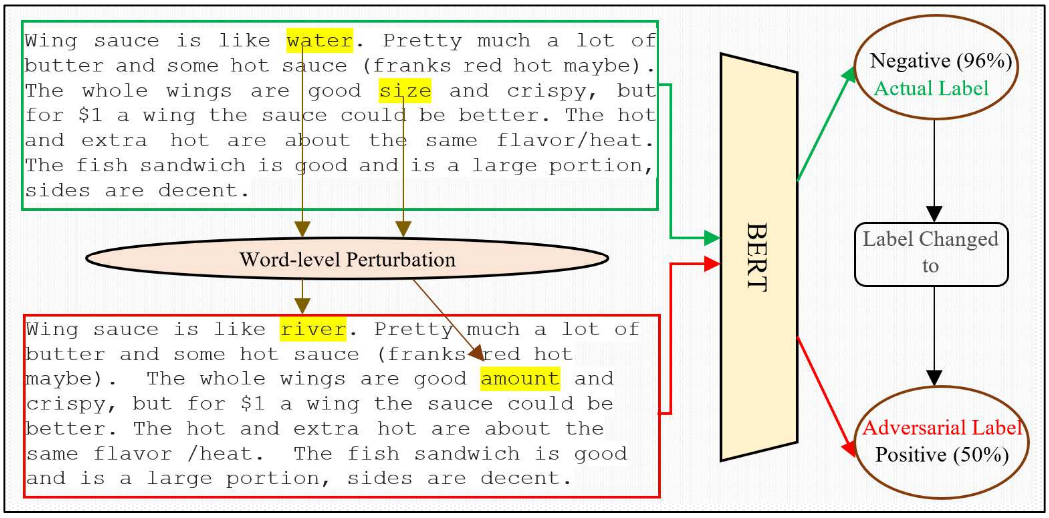
DNN:

The mapping function $\boldsymbol{F}: \boldsymbol{X} \rightarrow \boldsymbol{Y}$ from an input set $\boldsymbol{X}$ to the ground truth set $\boldsymbol{Y}$ may be employed to describe a conventional DNN. $\boldsymbol{Y}$ is a group of $\boldsymbol{k}$ labels such as *1, 2..., k*. $\boldsymbol{F}$ successfully classifies an input $\boldsymbol{x} \in \boldsymbol{X}$ to the actual label y, i.e., $\boldsymbol{F(x)} = \boldsymbol{y}$.

Perturbed example (Adversarial Input):

An adversary seeks to introduce a minor perturbation $\boldsymbol{\varepsilon}$ into $\boldsymbol{x}$ in order to create adversarial instance $\boldsymbol{x'}$, such that $F(\boldsymbol{x'}) = y'(\boldsymbol{y} \neq \boldsymbol{y'})$. Simultaneously, $\boldsymbol{x'}$ must not only deceive F, but it must also be undetectable to humans. In order to ensure that the generated $\boldsymbol{x'}$ is undetectable, a number of measures (such as similar resemblance in meaning) are used to attain this objective, i.e., $\|\boldsymbol{\varepsilon}\| < \boldsymbol{\delta}$. Here,$\boldsymbol{\delta}$ is a limiting threshold for the number of manipulations.

The field of computer vision (CV) has paid close observation to adversarial examples, which were initially proposed for image classification, in the past few years. The characteristics of hostile image instances have been extensively researched and investigated. The revelation that these assaults can be detected & resisted through randomization has resulted in a slew of promising follow-up studies, such as the creation of more resilient adaptive assaults and a better knowledge of adversarial scenario characteristics. Because languages are discrete [9], adversarial perturbed instances in textual processing assignments differ significantly from their counterpart in computer vision, and the features of textual adversarial cases have not been well explored. Comprehensive tests are conducted in this study to establish whether textual adversarial scenarios have similar characteristics[10]. We focus on text categorization and interpretation, among the most important and well-studied problems in the domain of adversarial attacks [11]. Pre-trained language models, which are a typical component of Language models, are of particular interest to us. Our findings would have a greater influence on the NLP community this way.



Wing sauce is like water. Pretty much a lot of butter and some hot sauce (franks red hot maybe). The whole wings are good size and crispy, but for $1 a wing the sauce could be better. The hot and extra hot are about the same flavor/heat. The fish sandwich is good and is a large portion, sides are decent.

Wing sauce is like river. Pretty much a lot of butter and some hot sauce (franks red hot maybe). The whole wings are good amount and crispy, but for $1 a wing the sauce could be better. The hot and extra hot are about the same flavor /heat. The fish sandwich is good and is a large portion, sides are decent.

Figure 3. Adversarial instance developed by manipulating *word*[4] in a manner such that it looks semantically similar to human but spoof BERT model in giving inaccurate prediction

Based on the perturbation units used in building adversarial instances, adversarial attacks in the NLP field can be characterised as sentence, word, character, or multi-stage. As visible from **Figure 2** char-level attacks imply that attackers manipulate many letters in words in order to create hostile samples which can mislead detectors [12]. Majority of the changes includes spelling errors, addition, exchange, removal, and reversal among the most prevalent operations. Word-level attacks [4] consist of a variety of word alterations as can be seen from **Figure 3**. In numerous ways, attackers construct hostile examples by inserting, altering, or removing specific phrases. Sentence-level attacks often include the insertion or rewriting of a sentence while preserving its meaning. Multi-level attacks integrate various perturbation attacks to create a stealthy and extremely effective attack. **Figure 4** depicts the classification of perturbation level in generating adversarial text.

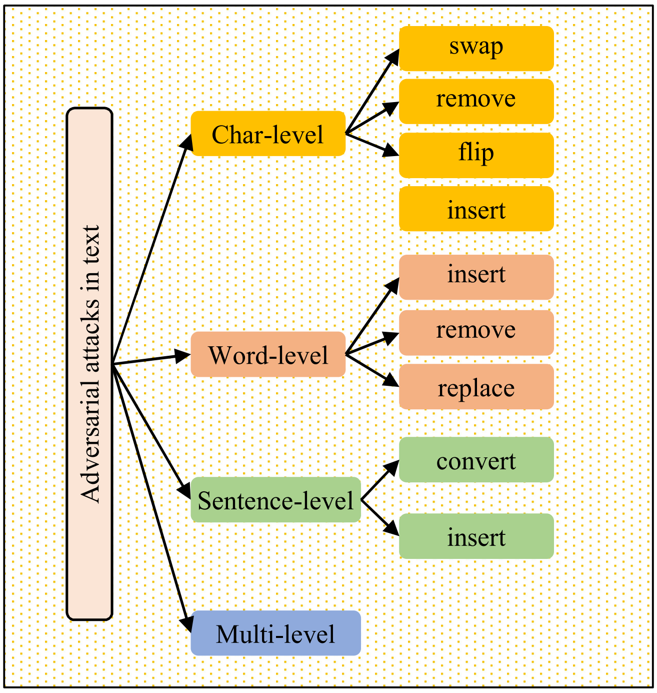


Figure 4 Classification of Adversarial attacks in text on basis of perturbation units

Each adversarial attack consists of four basic elements, including a transformation, a search strategy, a set of restraints as well as a target function, as outlined in the following section.

- *Modifications*: A modification that, given a single input, generates multiple potential perturbations. thesaurus word swap, word embedding word exchange Character substitution, are examples.
- *Search method:* A technique that checks the model on a regular basis and finds potential perturbations from a wide range of modifications. Particle swarm optimization search, genetic algorithm, beam search, and greedy search techniques ranking importance of different words in an input sequence are examples.
- *Restrictions:* A set of limitations used to evaluate the reliability of alterations in comparison to the real input. Instances include minimum language encoding cosine resemblance, word-embedding proximity, part-of-speech coherence and grammar checker.
- *Specificity:* A task-specific objective function that evaluates an attack's efficacy based on model outcomes. Exemplifications: untargeted and targeted classification, output that is not overlapping.

C. Attacks Recipes

The attack strategies were implemented on the Yelp polarity dataset to generate adversarial examples. These hostile samples are then utilized to alter the 3 aforementioned classifiers to categorize the positive sentiment of a business analytics review as negative and vice versa, so obscuring the detection of sentiment. A concise description of all attack strategies listed in **Table 1**. As depicted in **Figure 4**, these attack tactics are developed by perturbing the input text sequence at several levels, including sentence-level, word-level, and character-level.

Table 1 Adversarial Attack Recipes

Attack	Description
Text-bugger[12]	These attacks were tuned for use with apps from the real world. They make use of space insertions, character deletions, and character switching. In addition, they replace characters with similar-appearing letters (e.g., o with 0) and in context-aware language dimensional space, words with the topmost closest neighbors are used.
Text-Fooler[4]	The attack approach seeks exchange key words of an input sentence with their 50 nearest embedding neighbours. BERT optimisation.
PWWS [13]	The attack seeks to exchange key words in an input sequence with best suitable synonyms based on the saliency score of the term under a set of linguistic constraints.
PSO [14]	Attack at the word level using a sememe-based word replacement technique & particle-swarm-optimization.
Pruthi [11]	This attack method relies on character shifting, deletion, and insertion in the crucial words of an input sequence to preserve the semantic similarity of the sentence employed in this method.
Kuleshov[15]	substituting the important words in an input sentence from counter-fitted word embedding space under a set of semantic similarity and grammatical checks.
IGA [8]	Implemented as a way of defence against hostile attacks. Utilizes a mismatched word embedding swap.
DWB[5]	Tiny text perturbations are generated in a black-box environment. It employs several character swaps (swapping, substituting, removing, and inserting) with replace-1 scoring.
A2T[9]	This attack method employs gradient-based synonym word interchange in white-box hostile scenarios. Combining sentence encoding cosine similarity with grammatical checks, it maintains semantic similarity.
BAE [3]	Uses a BERT disguised language model transformation. It leverages the language model for token exchange so that it fits the context optimally.
Check-List[16]	Motivated by the fundamentals of behavioural testing. Modifies names, numerals, and localities, as well as use contractions and extensions.

III. Proposed Approach

We divided the investigation into 2 segments to assess the limitations of models designed to classify the sentiment of business reviews. Utilizing the "Yelp polarity" sentiment prediction dataset, advanced deep-learning classifiers were developed, and adversarial techniques were then used to change the pre-trained models' output. By perturbing the input, the objective of the attack algorithms is to deceive the model into making inaccurate predictions. To determine the efficacy of the attack models, we select 100 examples from the test set that are accurately classified, so that the accuracy of the classifiers does not influence the evaluation. These source texts are then fed into attack methodologies to develop adversarial samples. The adversarial samples are then sent to the classifier to generate the final prediction. The success rate of the assault algorithm is the proportion of incorrect predictions made by the classifier. A greater success rate indicates that the attacking algorithm may generate more formidable adversaries capable of causing the classifier to perform improperly. **Figure 5** depicts the methodology for conducting an adversarial approach on sentiment classification models. We evaluate 100 test samples and their percentage of altered words to calculate the attack success rate ASR (proportion of successfully attack instances to the sum of both successful and unsuccessful instances) for each attack method for all 3 models. The prospective outcomes of the conducted approach outcomes are depicted in **Table 3** for adversarial attack output for word-CNN, **Table 4** for word-LSTM and **Table 5** for BERT model. In this situation, a successful attack implies that the adversarial input can erroneously predict with high confidence. In the case of a failed attack, the adversarial sample is incapable of misclassifying the real prediction. The *Related work* section of the paper describes the different adversarial attack methodologies utilized against the victim models. Our final goal is to discover which model is more sensitive to adversarial manipulations and which is more durable, thus we calculate the average ASR for each model.

IV. Experimental Settings

This part outlines the description of the dataset used and victim models utilized for the investigation, along with the appropriate parameters for our experiment analysis. Then, these trained classifiers were assaulted by utilizing several cutting-edge adversarial attack algorithms.

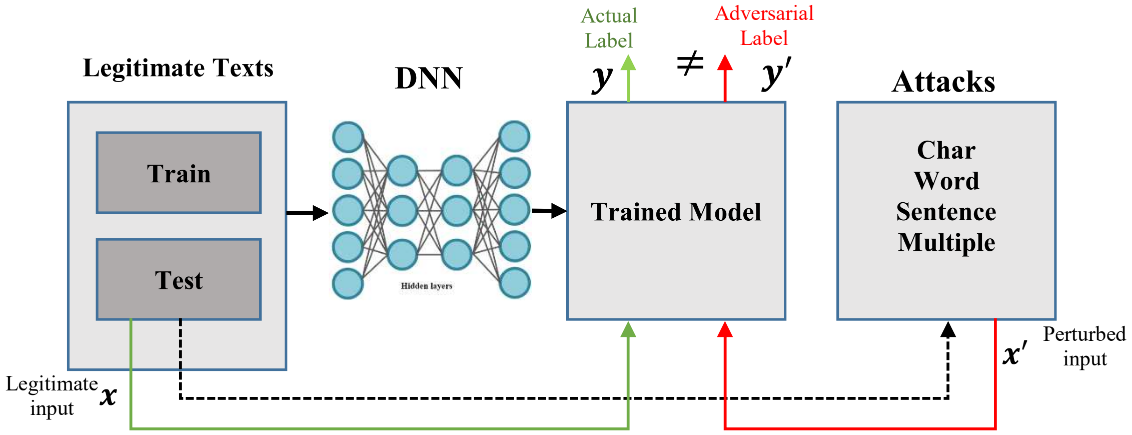


Figure 5 Framework for proposed methodology for conducting Adversarial Attack on Sentiment Classifiers

A. Dataset Description

The "Yelp polarity" dataset was utilized for this work. This is a binary sentiment analysis dataset. The dataset is created by assuming that stars 1 & 2 are negative and stars 3 & 4 are assigned positive. 280,000 training samples and 19,000 testing samples are drawn at random for each polarity. In total, there are 560,000 training samples and 38,000 test samples. Class 1 denotes negative polarity, while Class 2 denotes positive polarity. **Table 2** displays the classifiers trained on this dataset, along with their details.

B. Victim Models

The specifications and accuracy scores of the models trained on Yelp polarity binary sentiment classification dataset are shown in **Table 2** beneath.

Table 2 Description of different sentiment classifiers

Model	Parameterization	Acc. Scores
Word-CNN	For the Word-CNN model, we utilised three window sizes: 3, 4, and 5, with 100 filters for each window size and a dropout of 0.3. Use a 200-dimensional GLoVE embeddings basis.	**91.30%**
Word-LSTM	We employed a 1-layer bidirectional LSTM with 150 hidden neurons and a dropout of 0.3 for the Word-LSTM. Glove word embeddings trained on 6B tokens from Giga-words & Wikipedia were utilized.	**92.20%**
BERT	On the Yelp polarity dataset, the model bert-base-uncased was fine-tuned for sentiment classification. During the course of 5 epochs, the model was fine-tuned using a batch size of 16, a learning rate of 5e-05, and a maximum sequence length of 256. Because this was a classification assignment, the model was trained using a cross-entropy loss function. After four iterations, the model's greatest performance on this problem, as measured by the eval set confidence score, was *96.99%.*	**96.99%**

C. Attacking Victim Models

Table 3 Attack outcomes for model Word-CNN

Attack Method	Success (S)%	Failed (F)%	Skipped %	ASR= $(\frac{S}{S+F})$%	Average Perturbed Word%
A2T [9]	43	46	11	48.31	06.25
BAE [3]	68	21	11	76.40	04.28
Checklist [16]	1	88	11	01.12	42.75
DWB [5]	87	2	11	97.75	07.52
IGA [8]	68	21	11	76.40	14.60
Kuleshov [15]	84	5	11	94.44	11.42
PSO [14]	70	19	11	78.65	16.08
Pwws [13]	87	2	11	97.75	04.37
Pruthi [11]	12	77	11	13.48	03.44
Textbugger [12]	83	6	11	93.26	14.18
TextFooler [4]	88	1	11	98.87	05.04

Table 4 Attack outcome for Word-LSTM

Attack Method	Success (S)%	Failed (F)%	Skipped %	ASR= $(\frac{S}{S+F})$%	Average Perturbed Word%
A2T [9]	59	31	10	65.56	06.16
BAE [3]	79	11	10	87.78	04.39
Checklist [16]	3	87	10	03.33	46.98
DWB [5]	88	2	10	97.78	07.34
IGA [8]	69	21	10	76.66	15.90
Kuleshov [15]	85	5	10	94.44	10.02
PSO [14]	71	19	10	78.88	18.06
Pwws [13]	68	22	10	75.55	16.30
Pruthi [11]	10	80	10	11.11	02.58
Textbugger [12]	86	4	10	95.56	14.66
TextFooler [4]	89	1	10	98.88	04.85

Table 5 Attack outcome for BERT

Attack Method	Success (S)%	Failed (F)%	Skipped %	ASR= $\left(\frac{S}{S+F}\right)$ %	Average Perturbed Word%
A2T [9]	39	60	1	39.39	06.08
BAE [3]	64	34	1	64.64	05.43
Checklist [16]	02	97	1	02.02	45.66
DWB [5]	65	34	1	65.00	12.54
IGA [8]	71	28	1	71.71	14.44
Kuleshov [15]	89	10	1	89.89	11.22
PSO [14]	72	27	1	72.72	17.70
Pwws [13]	92	7	1	92.92	07.63
Pruthi [11]	13	86	1	13.13	09.64
Textbugger [12]	77	22	1	77.77	25.39
TextFooler [4]	91	8	1	91.91	10.32

V. Results & Discussion

The accuracy of the BERT framework is evidently superior to that of the word-CNN and word-LSTM models, as shown in **Table 2**. BERT is regarded for being the most efficacious transformer model; it demonstrates that transformer models are more accurate and efficient than former baselines. Afterwards, on each targeted model, the attack success rate of each assault is examined to determine which model is the most and least vulnerable. Using the Equation **(1)** below, the mean ASR on each of the model is calculated.

$$.\boldsymbol{S_r} = \frac{\sum_{i=1}^{a} \frac{S_i}{S_i+F_i}}{a} \qquad \textbf{(1)}$$

$$\boldsymbol{S_r} = Attack\ Success\ rate; \boldsymbol{a} = number\ of\ attack\ methods; \boldsymbol{S_i} = successful\ attack; \boldsymbol{F_i} = Failed\ attack$$

(As the skipped statements are based on model training rather than Attacks, they were removed from the evaluation. Examining the skipped values reveals the erroneous inputs the model initially predicted throughout its training)

Table 6 ASR results on the 3 Sentiment Classifiers

Model	Success Rate%
BERT	61.90
Word-LSTM	71.36
Word-CNN	70.54
Total Average	67.93

The results of an in-depth analysis of the data are presented in **Table 6**, which demonstrates that not only does the BERT model have the maximum confidence score in contrast to the other models, but it is also the model that is the least susceptible to being attacked by an adversary. Of the three models, the word-LSTM framework is the one that is most vulnerable to the effects of adversarial perturbations. We found that the overall success percentage of the attacks was

67.93% across all three types of sentiment classifiers. The Because the model BERT has the highest confidence score of ***96.99%*** and the lowest ASR of ***61.90%*** among Word-LSTM and Word-CNN, the BERT model is recognized as being the most powerful transformer model. This is due to the fact that it has the lowest ASR. With an accuracy score of ***92.20%*** for the Word-LSTM model and ***91.30%*** for the Word-CNN model, respectively, and ASR values of ***71.36%*** for the Word-LSTM model and ***70.54%*** for the Word-CNN model. It is abundantly evident that the model Word-LSTM is extremely sensitive to variations in the text input. According to the results of our comprehensive evaluation, the BERT model is more reliable than the other two models, hence it ought to be used whenever there is a possibility of adversarial manipulations. Also, the model was given the maximum possible accuracy score when evaluated using the Yelp polarity dataset. A plausible explanation for why the BERT model is performing better in terms of robustness and efficiency is that the "bert-base-uncased" model has 12 Layers, 110M parameters, 768 Equal and Hidden Embedding Layers, and 768 Hidden Layers. Here, we have used the "bert-base-uncased" version, which was optimised for the sentiment categorization task on the Yelp polarity dataset. Because of its size, its training requires a significant amount of computational power. Because it is a heavy model, however, it is least intuitive to malicious manipulations than Word-LSTM and Word-CNN models are. According to the findings of this research, the BERT model should be selected by future researchers and by those who use their sentiment categorization assignments for a variety of applications. This is because the BERT model is less vulnerable to hostile perturbations and produces higher confidence scores.

Table 7 Mean success rate of a specific attack method across all classifiers

Attack Method	**Success %**	**Perturbation Level**	**Average Perturbed Word%**
TextFooler [4]	96.55	word	06.73
Kuleshov [15]	92.92	word	10.88
Textbugger [12]	88.86	word & character	18.07
Pwws [13]	88.74	word	09.43
DWB [5]	86.84	character	09.13
PSO [14]	76.75	word	17.28
BAE [3]	76.27	word & character	04.70
IGA [8]	74.92	word	14.98
A2T [9]	51.08	word	06.16
Pruthi [11]	12.57	character	12.57
Checklist [16]	02.15	word	45.10

TextFooler[4] and Kuleshov et al.[15], both of which are word-perturbation strategies, come in at number one and number two, respectively, in **Table 7**'s assessment of the attacks to which models are most vulnerable. This demonstrates quite clearly that word-level perturbations are the ones that are most successful in tricking the systems. Word-level attacks have a greater propensity to have an effect on models compared to char-level or hybrid attacks (word & character mixed).

VI. Conclusion

The intent of this research was to offer a possible response to the query, "How susceptible is text-based sentiment categorization to adversarial threats?" This was evaluated by assessing if most of adversarial attack algorithms incorrectly identify the correct estimate of review sentiment. The findings show that text classification of sentiment prediction from business assessments can be evaded by easily altering the terms and letters used by the models of deep-learning and at the same time maintaining human observers' semantic similarity. Overall, it has been demonstrated that adversarial alterations can be used to obscure autonomous sentiment classification methods. For the sake of society as a whole, adversarial robust generalizations must therefore replace conventional generalizations. This paper encourages readers to seek out classifiers which are more resistant to adversarial manipulations in addition to those with higher confidence scores.

VII. Refrences

[1] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015, pp. 6562–6572.

[2] A. Bajaj and D. K. Vishwakarma, “Exposing the Vulnerabilities of Deep Learning Models in News Classification,” in *2023 4th International Conference on Innovative Trends in Information Technology (ICITIIT)*, IEEE, Feb. 2023, pp. 1–5. doi: 10.1109/ICITIIT57246.2023.10068577.

[3] S. Garg and G. Ramakrishnan, “BAE: BERT-based adversarial examples for text classification,” in *EMNLP 2020 - 2020 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 2020. doi: 10.18653/v1/2020.emnlp-main.498.

[4] D. Jin, Z. Jin, J. T. Zhou, and P. Szolovits, “Is BERT Really Robust? A Strong Baseline for Natural Language Attack on Text Classification and Entailment,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, Jul. 2019, pp. 8018–8025. [Online]. Available: http://arxiv.org/abs/1907.11932

[5] J. Gao, J. Lanchantin, M. Lou Soffa, and Y. Qi, “Black-box generation of adversarial text sequences to evade deep learning classifiers,” in *Proceedings - 2018 IEEE Symposium on Security and Privacy Workshops, SPW 2018*, 2018, pp. 1–21. doi: 10.1109/SPW.2018.00016.

[6] X. Han, Y. Zhang, W. Wang, and B. Wang, “Text Adversarial Attacks and Defenses: Issues, Taxonomy, and Perspectives,” *Security and Communication Networks*, vol. 2022. Hindawi Limited, 2022. doi: 10.1155/2022/6458488.

[7] A. Pandey and D. K. Vishwakarma, “Attention-based Model for Multi-modal sentiment recognition using Text-Image Pairs,” in *2023 4th International Conference on Innovative Trends in Information Technology (ICITIIT)*, IEEE, Feb. 2023, pp. 1–5. doi: 10.1109/ICITIIT57246.2023.10068626.

[8] X. Wang, H. Jin, Y. Yang, and K. He, “Natural Language Adversarial Defense through Synonym Encoding,” in *37th Conference on Uncertainty in Artificial Intelligence, UAI 2021*, 2021.

[9] J. Y. Yoo and Y. Qi, “Towards Improving Adversarial Training of NLP Models,” in *Findings of the Association for Computational Linguistics, Findings of ACL: EMNLP 2021*, 2021. doi: 10.18653/v1/2021.findings-emnlp.81.

[10] S. Qiu, Q. Liu, S. Zhou, and W. Huang, “Adversarial attack and defense technologies in natural language processing: A survey,” *Neurocomputing*, 2022.

[11] D. Pruthi, B. Dhingra, and Z. C. Lipton, “Combating adversarial misspellings with robust word recognition,” in *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, 2020. doi: 10.18653/v1/p19-1561.

[12] J. Li, S. Ji, T. Du, B. Li, and T. Wang, “TextBugger: Generating Adversarial Text Against Real-world Applications,” in *26th Annual Network and Distributed System Security Symposium*, 2019, pp. 1–15. doi: 10.14722/ndss.2019.23138.

[13] S. Ren, Y. Deng, K. He, and W. Che, “Generating natural language adversarial examples through probability weighted word saliency,” in *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*, 2020. doi: 10.18653/v1/p19-1103.

[14] Y. Zang *et al.*, “Word-level Textual Adversarial Attacking as Combinatorial Optimization,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 6067–6080. doi: 10.18653/v1/2020.acl-main.540.

[15] V. Kuleshov, S. Thakoor, T. Lau, and S. Ermon, “Adversarial Examples for Natural Language Classification Problems,” in *ICLR 2018 : International Conference on Learning Representations*, 2018.

[16] M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh, “Beyond Accuracy: Behavioral Testing of NLP models with CheckList,” *ACL 2020 - 58th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, pp. 4902–4912, 2020.