Group 3: Adversarial Techniques in NLP

Abhishek Krishna Deeksha Arora 11 Preeti Singh
Sambhrant Maurya Shruti Sharma
20111002, 20111017, 20111044, 20111054, 20111061
{krishnacs20, deeksha20, preeti20, samaurya20, shruti20}@iitk.ac.in
Indian Institute of Technology Kanpur (IIT Kanpur)

May 19, 2021

Abstract

As more and more AI systems are developed, the risk and the need for their security also grows manifold. Machine Learning systems are vulnerable to numerous real-life mischiefs. Recent attack methods on these models have exposed their vulnerability to certain adversarial inputs. Research in Adversarial Attacks in NLP has seen a boom in the past 4 years. In this survey report, we give a brief overview of the recent work done in this field. We also propose a novel idea of employing the recently developed Misspelling Oblivious Embeddings [1] for generating robust word replacements.

1 Introduction

With the recent advancements in technology, much development has been done for deploying Machine Learning and Deep Learning models in relevant fields. Such systems have been observed to outperform even their human counterparts. However, all these systems have exposed their over-confident and foolish attitude towards adversarial (manipulated) input data [2][3] which highlights the lack of perception in the algorithms.
An *adversarial example* is a slight change in the input of a machine/deep learning model, which causes the model to predict incorrect labels. It is simpler to generate such examples in image processing/computer vision; however, in Natural Language Processing, depending on the context, an adversarial example should be grammatical and semantically similar to original input. This motivated us to work in this field and explore the underlying ideas of different types of attacks related to NLP. Adversarial techniques have applications in Sentence/Text classification, Malware detection, Machine comprehension, Machine Translation, Medical records, Paraphrasing, etc.
Adversarial attacks can be categorized as follows[4][5]:

1. Based on model accessibility, it can be a **white-box attack** or a **black-box attack**. In the former, the attacker has access to model parameters, but in black-box setting, the attacker has no information about the model.
2. Attack can be **targeted** or **untargeted**. In targeted attacks, the attacker wants the model to predict a specific target label, whereas in untargeted attacks, the attacker wants the model to predict any label different from the true label.
3. Based on Granularity, the attack can be at **character-level**, **word-level**, or **sentence-level**.
4. Attack can be **single modal** like on Dialogue Systems or **cross modal** like on Speech Recognition Systems.
5. Based on the type of Deep Neural Network being attacked, the attack can be on CNNs, RNNs, Hybrid models, etc.

Although the catch is to generate adversarial examples in a cost-sensitive and speedy manner such that the model's robustness is tested, these adversarial examples can also be used to re-train the model to further improve the generalization and robustness of the model. This report summarizes various

kinds of attacks performed on Natural Language Processing systems with a focus on text classifiers and machine comprehension systems. We also compile some adversarial text generation techniques in literature and provide a comprehensive comparison of them. In the end, we propose our novel idea that can possibly generate robust adversarial examples.

2 Approaches for Adversarial Attacks in NLP

2.1 Gradient based approaches

Early work in this direction was done by Nicolas Papernot(2016)[6] who adapted cost based gradient approach(FGSM) and forward derivative to craft adversarial sequences for misleading categorical and sequential RNN. The idea was to use forward derivative to unfold the computational graph then compute the Jacobian which can be used to craft adversarial examples by considering the Jacobian's $J_f[:, j]$ column corresponding to one of the output components j.

Bin Liang et.al (2018)[7] used backpropagation to compute the cost gradients $\nabla_x J(F, x, c)$ for every training example x to get the cost gradients of every dimension in all character vectors of these samples. The characters containing the dimensions with the highest magnitude were selected to be perturbed. In the same year, Carlini et.al.(2018)[8] constructed a targeted audio adversarial attack on speech-to-text translation models using iterative optimization based techniques in a white box setting. The idea behind the attack is to add a nearly inaudible perturbation δ to a natural waveform to get the desired phrase. Ebrahimi et.al.(2018)[9] also proposed a white-box adversarial attack(HotFlip) which generates adversarial examples through atomic flips using directional derivative and uses beam search to find the best direction for multiple flips. Another work by Ebrahimi et.al. (2018)[10]has adapted the idea of HotFlip to attack neural machine translation models and also introduced controlled and targeted adversaries, where the controlled adversaries mutes the chosen words in the original translation and targeted adversaries replace the chosen words. Zhao et.al.(2018)[11] proposed a framework to generate natural adversarial text in semantic space by training WGAN and Inverter and minimizing the reconstruction error. By searching for samples in a latent low-dimensional space z using either iterative stochastic search or hybrid shrinking search and then mapping it to a space x to identify adversaries, the generated adversaries were valid and semantically close to the input.

In 2019, Yotam Gil[12] proposed an idea where they produced adversarial examples using a white-box attack (HotFlip), and trained a neural network model with these examples to imitate a white-box attack. They also used this trained model to perform a black-box attack which demonstrated high-speed generation of adversarial examples. Later that year, Behjati et.al.(2019)[13] proposed a universal attack framework using gradient projection. The attack uses an iterative projected gradient-based approach to find a sequence of words that can be inserted into the input sequences. Wallace et al.(2019)[14] also showed that text classifiers are vulnerable to universal adversarial attacks. However, those word sequences were unnatural, meaningless, and could be easily distinguished from natural text. Yi-Ting Tsai(2019)[15] proposed a white-box algorithm using global search. The idea was to use gradient of an objective function to find perturbations and to update these perturbations via backpropagation. They proposed to use two regularization terms in the objective function, one for large perturbations and other for large distances, which tries to maximize the difference between the sigmoid value of the input and the perturbed input. Li et al. (2019)[16] devised TextBugger to generate adversarial examples and evaluated it on sentiment analysis and toxic content detection systems. In the white-box setting, it uses a Jacobian matrix of input to identify important words for replacement. TextBugger achieved 95.2% success rate on IMDB dataset on the LR model.

In 2020, Song et.al.[17] proposed Natural Universal Trigger Search (NUTS) algorithm which leverages an Adversarially Regularized AutoEncoder (ARAE) to generate triggers and performs gradient-based search over target triggers to output natural text that fools a target classifier. Zhang et.al.(2020)[18] used Metropolis-Hastings Sampling(Metropolis et.al.(1953)[19]) to generate fluent adversarial examples given a stationary distribution and a transition proposal, where they incorporate gradients for the generation of the pre-selection score which in turn generates the candidate set for a chosen word. To evaluate robustness of sequence-to-sequence models, Cheng et.al.(2020)[20] proposed an optimisation

based framework - Seq2Sick, wherein the issue of discrete input space of texts is solved by projected gradient descent and group lasso regularization to enforce sparsity of distortion. Seq2Sick achieves success rate of 84% to 100% on text summarization and machine translation. In this framework, two attacks are performed by optimising hinge-like loss function on a logit layer - non-overlapping attack and targeted keyword attack.

2.2 Human-in-the-loop approaches

Wallace et.al.(2019)[21] argued that automated generation of adversarial examples exposes only the superficial patterns in text and hence these methods are limited in complexity and diversity. For exploring more complex failure patterns, they proposed a human-in-the-loop generation framework where trivia enthusiasts (humans who write questions for academic competitions) are guided via a GUI to craft diverse adversarial examples that break existing QnA models. Zhou et.al.(2019)[22] also argued that the weakness of state of the art fake news detector models is that they rely on linguistic characteristics of an article to verify its credibility without performing fact checking, making them vulnerable to fact-tampering attacks. They generated adversarial examples by hand from real news in the McIntire's dataset and showed that state of the art fake news detector models are vulnerable to fact distortion, subject-object exchange and cause confounding attacks.

In 2020, Kashabi et.al.[23] created large training sets, optimally, for training deep neural networks. They naturally perturbed BoolQ[24] to produce 17K questions using 4K seed questions. Crowdworkers construct clusters of minimally-perturbed samples during training with mixed labels by keeping perturbation cost ratio low. The accuracy of RoBERTA model was shown to drop by 15%. For evaluating RC-based QA on SQuAD[25] dataset, Rahurkar et.al. (2020)[26] perturbed the background context, by introducing both word-level and sentence-level perturbations without changing the question. They showed that attributions (word importance) can improve the attacks proposed by Jia and Liang [27].

2.3 Semantic Similarity Based Approaches

In 2018, Sato et.al.[28] formulated the interpretable AdvT-Text or iAdvT-Text algorithm to generate interpretable adversarial examples by limiting the directions of perturbations towards existing words in the input word embedding space. Yanjun Qi(2018)[29] presented an algorithm - DeepWordBug which performs character-level transformation in a black-box setting. The authors had introduced two scores: Temporal score(TS) and Temporal Tail Score(TTS) - which scores a word based on its preceding and succeeding word. Temporal scores can be computed using one forward pass of RNN and TTS is a complement of TS. These scores are used to rank the words for perturbation. Yang et.al., (2018)[30] developed Greedy attack, which is a perturbation based method and Gumbel attack, which is a scalable learning method to perform adversarial attacks on discrete data. Greedy attack chooses a replacement word such that it is closest to the original word in terms of Euclidean distance while Gumbel attack learns a parametric sampling distribution for generating perturbations.

Ren et.al.(2019)[31] proposed a new method, PWWS to substitute a word based on synonyms strategy using word saliency and classification probability. They replaced words with their synonyms and name entities(NEs) with similar NEs. They ranked all the probable substituted synonyms and then, substituted the words one by one. Models like Bi-LSTM, CNN on IMDB, AG's News datasets experience a drastic drop in accuracy(86% to 5.5%) when attacked using PWWS. Liang et.al. (2019)[32] proposed a generator and classifier based method. The generator is used to generate adversarial examples by replacing words of input text with their synonyms. If the generated example successfully confuses the classifier, it is given a high reward and it is used to train the generator in a reinforcement learning setting.

Morris et.al.(2020)[33] proposed four categories of constraints that an adversarial example should follow - semantics, grammar, overlap and unidentifiable to humans. They proposed a TFAdjusted attack which resembles TextFooler but with more requisite constraints. Jin et.al. (2020)[34] used synonym replacement strategy for word replacements using word embeddings from Mrksic et.al (2016)[35], where cosine similarity is used as a measure of similarity between words.
Hsieh et.al.,(2020)[36] used random, list-based, greedy select+greedy replace, greedy select+embedding

constraint and attention based select attacks on LSTM, Bert and Transformers. For sentiment analysis, text entailment and MT, they showed that self-attention models have less success attack rates as compared to LSTM. Samson et al.. (2020)[37] proposed a new method based on inflectional morphology of the words. For each word in input text, if POS of the word is a noun, verb or an adjective, they greedily search inflected forms of these words and then find the form that caused the greatest increase in loss. Basemah et.al.(2020)[38] proposed a new approach to improve generalization of DNNs. The important words in the input text are replaced with the average embedding of their synonyms. They used the Replace-1 scoring function and showed that, on average, it increases the accuracies of CNN and Bi-LSTM models by 41.30% and 55.66%.

In a recent work, Sabir et.al.(2021)[39] have proposed a reinforcement learning framework, which learns a policy that will be transferable to unseen datasets and will generate utility preserving adversarial examples.

2.4 Sentence Paraphrasing

Sentence paraphrasing is a sentence level perturbation where chosen sentences are replaced with semantically similar sentences, but can fool the target model to mispredict.
Iyyer et.al.(2018)[40] trained general-purpose syntactically controlled paraphrase networks (SCPN) to paraphrase the sentences. SCPNs generate paraphrases by inputting a back-translated paraphrase and a target syntax into the decoder and getting target paraphrase back from the decoder. The 20 most frequent templates from PARANMT-50M are selected as parse templates. Semantically inconsistent sentences are pruned using n-gram overlap and paraphrase similarity.

Gan et.al.(2019)[41] devised a novel method to generate paraphrased questions by training a transformer based model from Vaswani et al.(2017)[42] to output paraphrased questions given a source question and a paraphrased suggestion. They also created an adversarial paraphrased test set by re-writing the original question using words in the context near a confusing answer candidate of the same type as the correct answer. The performance of three state of the art QA models- BERT (Devlin et.al.(2018)[43]), DrQA4 (Chen et.al.(2017)[44]), and BiDAF5 (Seo et.al.(2016)[45]) dropped on both the paraphrased test sets. Zhang et.al. (2019)[46] proposed a new dataset, PAWS which contains paraphrase and non-paraphrase pairs that have high BOW representation. Word swapping is done by a CRF-based POS tagger. Using Beam search, based on input tag, candidate words are drawn and using english-german Machine Translation, it generates top-k german translations and translates them back to english. Models like BERT gained significant accuracy when trained using PAWS training examples.

Wu et.al.(2020)[47] employed sentence paraphrasing for targeting Machine Comprehension models where they paraphrase each sentence using ParaBank Rewriter (Hu et al., 2019) which uses Machine Translation for paraphrasing. Chenglei et.al.(2020)[48] adopted the Syntactically Controlled Paraphrase Network (SCPN) (Iyyer et.al.(2018)[40]) to generate paraphrases of a sentence based on a given syntactic parsing template to target Machine Reading Comprehension models.

2.5 Leveraging Language Models

Some recent works have leveraged SOTA performance of Maskes Language Models (MLMs) to perform adversarial attacks. Garg et.al(2020)[49] and Li et.al(2020)[50] have employed a BERT-based MLM setting to generate word replacements for attacking BERT-based models where they mask the chosen word and use BERT to generate alternate words in the context of the sentence, and then choose the most appropriate replacement.

Recently, Modi et.al(2021)[51] have employed the idea of Occlusion and Language Models(OLM) to perform efficient word ranking where OLM is used to generate the OLM relevance score r by sampling some candidate instances for a word and then replacing that word. The relevance score r, given a word of the input x, x_i, the incomplete input without this word $x_{\setminus i}$, prediction function f, label y and the logit value f_y is given by:

$$r_{f,y}(x_i) = f_y(x_i) - f_y\left(x_{\setminus i}\right) \tag{1}$$

2.6 Antonym and Named-Entity Based Semantic Altering Approaches

Jia and Liang (2017)[27] evaluated reading comprehension systems using SQuAD[25] dataset. They attacked 16 SOTA deep-learning models and found them vulnerable to concatenation adversaries. AddSent, a black-box attack, appends grammatically similar sentences at the end of the paragraph, while AddAny assumes that the model returns a probability distribution over many answers instead of a single answer and attacks by inserting random sequences of d English words. However, the attacked models do well when the question has an exact n-gram match with the original paragraph.

AddSent based adversarial training failed because it allows the model to learn certain superficial assumptions like position of distractor statements or a certain set of answers which shouldn't be picked. Wang et al.(2018)[52] modified the AddSent algorithm to formulate AddSentDiverse algorithm which generates varied adversarial examples by randomizing the position of distractor statements, generating dynamic fake answers and performing semantic feature enhancement.

Instead of directly attacking RC based QA systems, Nizar and Kobren (2020)[53] did it in 2 steps. First, they approximate a victim black-box model via model extraction and then perform a non-targeted attack on the extracted model using AddAny-kBest. AddSent doesn't require model extraction step and the adversary chosen is one that reduces instance-level F1-score of the model from the candidates returned. AddAny-kBest terminates when all of the k-best spans s returned by extracted model have F1-score of 0,

$$\underset{s \in S_i^k}{max}\ F1(s, s^*) = 0 \quad \text{, s* is the ground truth span}$$

They observed that AddSent and AddOneSent attacks are less effective than AddAny-kBest approach. Shudi et.al.(2020)[54] proposed a new KBAA algorithm against Knowledge Based Question Answering (KBQA) tasks. They used 'entity, relationship, entity' triplet to represent relationships. From the input question, NER is used to find the entity and collects all the possible data sets containing this entity using domain knowledge and AM is used to match the corresponding attribute.

2.7 Benchmarking and Comparing Different NLP Approaches

Morris et.al.(2020)[55] introduced a new python open-source framework, TextAttack. TextAttack is designed to benchmark and compare different NLP attacks from literature on various datasets. An attack can be composed using a goal function, set of constraints, transformation and search method.
Zeng et.al.(2020)[56] proposed OpenAttack which is an open source textual adversarial attack toolkit written in python. It provides a platform to evaluate various attacks based on different perspectives and also provides more evaluation metrics than TextAttack.
Yong et.al.(2020)[57] tried to find out which search method should be used to generate adversarial examples. On Yelp, MR, SNLI datasets, results show that in terms of success rate, beam and particle swarm optimization are leading scores and under time constraint, greedy WIR is the preferable method.

2.8 Other Approaches

Gradient based methods for adversarial example generation don't work well in the field of Natural Language Processing since they depend on the fact that small perturbations in the original text go unnoticeable by human viewers. Alzantot et al.(2018)[58] performed population based optimization via genetic algorithms to generate syntactically and semantically similar adversarial examples.

Meng et.al.(2020)[59] iteratively approximated the decision boundary of Deep Neural Networks to generate adversarial text where they ranked the words on the basis of their saliency scores and then geometric information was used to find the best synonym to replace word w, where w is the word with highest saliency score. Most of the State Of The Art fake news detector models use article's content, users' comments and replies to classify it as real or fake. Le et al.(2020)[60] exploited this fact in their novel framework called MALCOM, to fool fake news detector models by generating malicious comments. They used a conditional sequential text generation model to generate comments which are realistic and relevant to the article.

3 Research Proposal and Future Work

Research proposal: Previous work on Word-level adversarial attacks has relied on word2vec and GLoVE[27][16] embeddings for vector representation of words and generating perturbations. Some recent works have explored BERT[50]. Word2vec and GLoVE often fail to yield embeddings for out-of-vocabulary (OOV) words, i.e. words that were unseen at training time. To address this issue, we propose to use the recently developed Misspelling Oblivious Embeddings[1] by Facebook, which is built on top of the open source library FastText, whose loss function aims to closely embed words that occur in the same context. This embedding also comes with an inbuilt cosine similarity function and we expect this embedding to give better word replacements than its counterparts. We intend to use the existing TextAttack framework to explore this embedding and report its performance against the previously used word embeddings.

Future work: Recent works in adversarial attacks on text have tried to incorporate Language Models into this field. While Language Models, especially Masked Language Models (MLMs) like BERT have succeeded in generating high quality adversarial samples with high success rates and minimum perturbation, MLMs like BERT sometimes generate antonyms, or irrelevant word replacements which are semantically incoherent with the original word. Hence, there is a need to impose a restriction on the similarity measure during word replacements. Broadly, there is a need to improve language models to generate more semantically relevant perturbations.

4 Conclusion

As NLP applications like Machine Translation, Machine Comprehension, Speech-to-Text models, etc. get deployed in real world applications, their security becomes a concern. The threat of adversarial attacks is real as causing such models to mispredict can cause severe real life issues. In 2016, Microsoft claimed that their AI tweet bot "Tay" was attacked, causing it to tweet racist content[61]. We have tried to integrate almost all the recent work done on textual adversarial attacks in section 2 of this report. As the research in this domain is relatively new, there is a lot to explore. With new models come newer vulnerabilities. We, as developers and researchers need to make sure that we are aware of these vulnerabilities before attackers get to exploit them.

5 Individual Contributions

Roll No.	Name	Number of papers contributed	Papers
20111002	Abhishek Krishna	10	[31],[32],[33],[36], [37],[38],[46],[54], [55][57]
20111017	Deeksha Arora	10	[8],[17],[22],[28], [30],[41],[52],[58], [59],[60]
20111044	Preeti Singh	10	[5],[6],[9],[10], [12],[15],[29],[39], [46],[56]
20111054	Sambrant Maurya	10	[7],[13],[18],[21], [34],[47],[48],[49], [50],[51]
20111061	Shruti Sharma	10	[4],[11],[16],[20], [23],[25],[26],[27], [40],[53]

References

[1] A. Piktus, N. B. Edizel, P. Bojanowski, E. Grave, R. Ferreira, and F. Silvestri, "Misspelling oblivious word embeddings," June 2019.

[2] J. S. Ian J Goodfellow and C. Szegedy, "Explaining and harnessing adversarial examples," in *3rd International Conference on Learning Representations*, 2015.

[3] I. S. Christian Szegedy, Wojciech Zaremba and J. Bruna, "Intriguing properties of neural networks," in *2nd International Conference on Learning Representations*, 2014.

[4] W. E. Zhang, Q. Z. Sheng, A. Alhazmi, and C. Li, "Adversarial attacks on deep learning models in natural language processing: A survey," 2019.

[5] W. Wang, L. Wang, R. Wang, Z. Wang, and A. Ye, "Towards a robust deep neural network in texts: A survey," 2020.

[6] N. Papernot, P. McDaniel, A. Swami, and R. Harang, "Crafting adversarial input sequences for recurrent neural networks," 2016.

[7] B. Liang, H. Li, M. Su, P. Bian, X. Li, and W. Shi, "Deep text classification can be fooled," *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, Jul 2018.

[8] N. Carlini and D. Wagner, "Audio adversarial examples: Targeted attacks on speech-to-text," 2018.

[9] J. Ebrahimi, A. Rao, D. Lowd, and D. Dou, "HotFlip: White-box adversarial examples for text classification," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, (Melbourne, Australia), pp. 31–36, Association for Computational Linguistics, July 2018.

[10] J. Ebrahimi, D. Lowd, and D. Dou, "On adversarial examples for character-level neural machine translation," in *Proceedings of the 27th International Conference on Computational Linguistics*, (Santa Fe, New Mexico, USA), pp. 653–663, Association for Computational Linguistics, Aug. 2018.

[11] Z. Zhao, D. Dua, and S. Singh, "Generating natural adversarial examples," 2018.

[12] Y. Gil, Y. Chai, O. Gorodissky, and J. Berant, "White-to-black: Efficient distillation of black-box adversarial attacks," June 2019.

[13] M. Behjati, S. Moosavi-Dezfooli, M. S. Baghshah, and P. Frossard, "Universal adversarial attacks on text classifiers," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 7345–7349, 2019.

[14] E. Wallace, S. Feng, N. Kandpal, M. Gardner, and S. Singh, "Universal adversarial triggers for attacking and analyzing nlp," 2021.

[15] Y.-T. Tsai, M.-C. Yang, and H.-Y. Chen, "Adversarial attack on sentiment classification," in *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, (Florence, Italy), pp. 233–240, Association for Computational Linguistics, Aug. 2019.

[16] J. Li, S. Ji, T. Du, B. Li, and T. Wang, "Textbugger: Generating adversarial text against real-world applications," Internet Society, 2019.

[17] L. Song, X. Yu, H.-T. Peng, and K. Narasimhan, "Universal adversarial attacks with natural triggers for text classification," 2021.

[18] H. Zhang, H. Zhou, N. Miao, and L. Li, "Generating fluent adversarial examples for natural languages," 2020.

[19] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *The Journal of Chemical Physics*, vol. 21, no. 6, pp. 1087–1092, 1953.

[20] M. Cheng, J. Yi, P.-Y. Chen, H. Zhang, and C.-J. Hsieh, "Seq2sick: Evaluating the robustness of sequence-to-sequence models with adversarial examples," 2020.

[21] E. Wallace, P. Rodriguez, S. Feng, I. Yamada, and J. Boyd-Graber, "Trick me if you can: Human-in-the-loop generation of adversarial examples for question answering," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 387–401, Mar. 2019.

[22] Z. Zhou, H. Guan, M. Bhat, and J. Hsu, "Fake news detection via nlp is vulnerable to adversarial attacks," *Proceedings of the 11th International Conference on Agents and Artificial Intelligence*, 2019.

[23] D. Khashabi, T. Khot, and A. Sabharwal, "More bang for your buck: Natural perturbation for robust question answering," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Nov. 2020.

[24] C. Clark, K. Lee, M.-W. Chang, T. Kwiatkowski, M. Collins, and K. Toutanova, "BoolQ: Exploring the surprising difficulty of natural yes/no questions," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, (Minneapolis, Minnesota), June 2019.

[25] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," 2016.

[26] "Human adversarial qa: Did the model understand the paragraph?,"

[27] R. Jia and P. Liang, "Adversarial examples for evaluating reading comprehension systems," in *2017 Conference on Empirical Methods in Natural Language Processing*, (Copenhagen, Denmark), 2017.

[28] M. Sato, J. Suzuki, H. Shindo, and Y. Matsumoto, "Interpretable adversarial perturbation in input embedding space for text," 2018.

[29] J. Gao, J. Lanchantin, M. L. Soffa, and Y. Qi, "Black-box generation of adversarial text sequences to evade deep learning classifiers," 2018.

[30] P. Yang, J. Chen, C.-J. Hsieh, J.-L. Wang, and M. I. Jordan, "Greedy attack and gumbel attack: Generating adversarial examples for discrete data," 2018.

[31] S. Ren, Y. Deng, K. He, and W. Che, "Generating natural language adversarial examples through probability weighted word saliency," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, (Florence, Italy), July 2019.

[32] J. Xu, L. Zhao, H. Yan, Q. Zeng, Y. Liang, and X. Sun, "LexicalAT: Lexical-based adversarial reinforcement training for robust sentiment classification," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, (Hong Kong, China), Nov. 2019.

[33] J. Morris, E. Lifland, J. Lanchantin, Y. Ji, and Y. Qi, "Reevaluating adversarial examples in natural language," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, Nov. 2020.

[34] 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," 2020.

[35] N. Mrkšić, D. Séaghdha, B. Thomson, M. Gašić, L. Rojas-Barahona, P.-H. Su, D. Vandyke, T.-H. Wen, and S. Young, "Counter-fitting word vectors to linguistic constraints," 2016.

[36] Y.-L. Hsieh, M. Cheng, D.-C. Juan, W. Wei, W.-L. Hsu, and C.-J. Hsieh, "On the robustness of self-attentive models," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, (Florence, Italy), July 2019.

[37] S. Tan, S. Joty, M.-Y. Kan, and R. Socher, "It's morphin' time! Combating linguistic discrimination with inflectional perturbations," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, (Online), July 2020.

[38] B. Alshemali and J. Kalita, "Generalization to mitigate synonym substitution attacks," in *Proceedings of Deep Learning Inside Out (DeeLIO): The First Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, Nov. 2020.

[39] B. Sabir, M. A. Babar, and R. Gaire, "Reinforcebug: A framework to generate adversarial textual examples," 2021.

[40] M. Iyyer, J. Wieting, K. Gimpel, and L. Zettlemoyer, "Adversarial example generation with syntactically controlled paraphrase networks," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, (New Orleans, Louisiana), June 2018.

[41] W. C. Gan and H. T. Ng, "Improving the robustness of question answering systems to question paraphrasing," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, (Florence, Italy), pp. 6065–6075, Association for Computational Linguistics, July 2019.

[42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[43] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2019.

[44] D. Chen, A. Fisch, J. Weston, and A. Bordes, "Reading wikipedia to answer open-domain questions," 2017.

[45] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi, "Bidirectional attention flow for machine comprehension," 2018.

[46] Y. Zhang, J. Baldridge, and L. He, "PAWS: Paraphrase adversaries from word scrambling," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, (Minneapolis, Minnesota), June 2019.

[47] W. Wu, D. Arendt, and S. Volkova, "Evaluating neural machine comprehension model robustness to noisy inputs and adversarial attacks," 2020.

[48] C. Si, Z. Yang, Y. Cui, W. Ma, T. Liu, and S. Wang, "Benchmarking robustness of machine reading comprehension models," 2020.

[49] S. Garg and G. Ramakrishnan, "BAE: BERT-based adversarial examples for text classification," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (Online), pp. 6174–6181, Association for Computational Linguistics, Nov. 2020.

[50] L. Li, R. Ma, Q. Guo, X. Xue, and X. Qiu, "BERT-ATTACK: Adversarial attack against BERT using BERT," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (Online), pp. 6193–6202, Association for Computational Linguistics, Nov. 2020.

[51] V. Malik, A. Bhat, and A. Modi, "Adv-olm: Generating textual adversaries via olm," 01 2021.

[52] Y. Wang and M. Bansal, "Robust machine comprehension models via adversarial training," June 2018.

[53] N. J. Nizar and A. Kobren, "Leveraging extracted model adversaries for improved black box attacks," 2020.

[54] S. Guo, S. Wang, B. Liu, and T. Shi, "KBAA: An adversarial example generation method for KBQA task," in *Proceedings of Seventh International Conference on Dependable Systems and Their Applications*, Nov. 2020.

[55] J. Morris, E. Lifland, J. Y. Yoo, J. Grigsby, D. Jin, and Y. Qi, "TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Oct. 2020.

[56] G. Zeng, F. Qi, Q. Zhou, T. Zhang, B. Hou, Y. Zang, Z. Liu, and M. Sun, "Openattack: An open-source textual adversarial attack toolkit," 2020.

[57] J. Y. Yoo, J. Morris, E. Lifland, and Y. Qi, "Searching for a search method: Benchmarking search algorithms for generating NLP adversarial examples," in *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, Nov. 2020.

[58] M. Alzantot, Y. Sharma, A. Elgohary, B.-J. Ho, M. Srivastava, and K.-W. Chang, "Generating natural language adversarial examples," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, (Brussels, Belgium), pp. 2890–2896, Association for Computational Linguistics, Oct.-Nov. 2018.

[59] Z. Meng and R. Wattenhofer, "A geometry-inspired attack for generating natural language adversarial examples," 2020.

[60] T. Le, S. Wang, and D. Lee, "Malcom: Generating malicious comments to attack neural fake news detection models," *2020 IEEE International Conference on Data Mining (ICDM)*, pp. 282–291, 2020.

[61] Paul, "The racist hijacking of microsoft's chatbot shows how the internet teems with hate," Mar 2016.