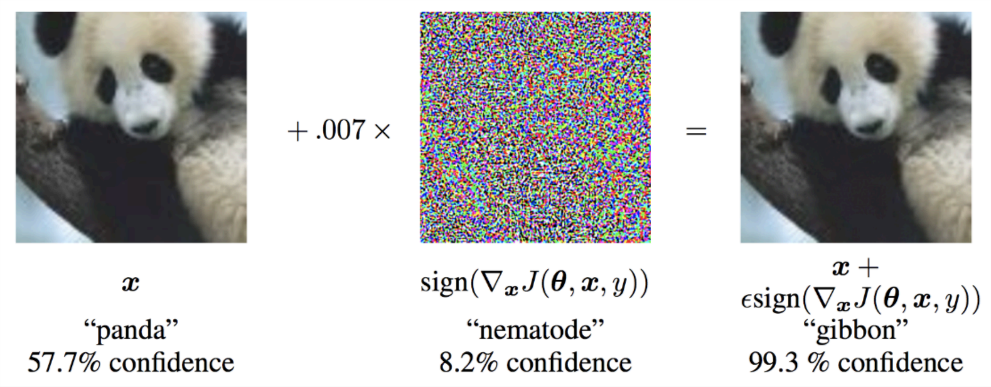
Generating Natural Language Adversarial Examples

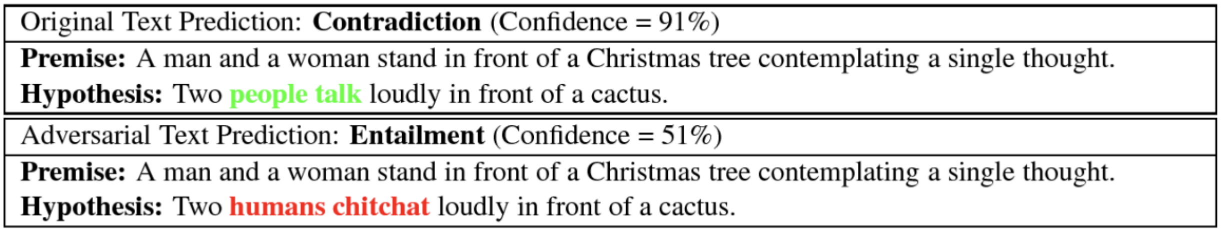
Blog Post by Shruti Sharan
UID: 405228029

Machine learning is exciting and becoming ubiquitous with every passing day. However, just like any new technology or invention, ML enables not only new amazing capabilities, but unfortunately, it also brings new vulnerabilities. One such vulnerability is known as Adversarial Attacks.

An evasion attack happens when the network is fed an “adversarial example”, ie, a carefully perturbed input that looks and feels exactly the same as its untampered copy to a human, but it completely throws off the classifier. Thus, it is a way to fool models through abnormal input (caused by perturbation).
An adversarial example is an input to a machine learning model that is intentionally designed by an attacker to fool the model into producing an incorrect output. For example, we might start with an image of a panda and add a small perturbation that has been calculated to make the image be recognized as a gibbon with high confidence.


$+ .007 \times$
$=$
$\boldsymbol{x}$
“panda”
57.7% confidence
$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
“nematode”
8.2% confidence
$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
“gibbon”
99.3 % confidence

Deep neural networks (DNNs) are vulnerable to adversarial examples, perturbations to correctly classified examples which can cause the model to misclassify. In the image domain, these perturbations are often virtually indistinguishable to human perception, causing humans and state-of-the-art models to disagree. However, in the natural language domain, small perturbations are clearly perceptible, and the replacement of a single word can drastically alter the semantics of the document.



Original Text Prediction: **Contradiction** (Confidence = 91%)

Premise: A man and a woman stand in front of a Christmas tree contemplating a single thought.
Hypothesis: Two **people talk** loudly in front of a cactus.

Adversarial Text Prediction: **Entailment** (Confidence = 51%)

Premise: A man and a woman stand in front of a Christmas tree contemplating a single thought.
Hypothesis: Two **humans chitchat** loudly in front of a cactus.

Eg of Adversarial Attack results against the textual entailment model. Modified words are highlighted in green and red for the original and adversarial texts respectively.

Adversarial examples have been generated through solving an optimization problem, attempting to induce misclassification while minimizing the perceptual distortion. Due to the computational cost of such approaches, fast methods were introduced which, either in one-step or iteratively, shift all pixels simultaneously until a distortion constraint is reached. Almost all popular methods are gradient-based. This approach obviously does not transfer to the natural language domain, as all changes are perceptible. A straightforward workaround is to project input sentences into a continuous space (e.g. word embeddings) and consider this as the model input. However, this approach also fails because it still assumes that replacing every word with words nearby in the embedding space will not be noticeable. Replacing words without accounting for syntactic coherence will certainly lead to improperly constructed sentences which will look odd to the reader.

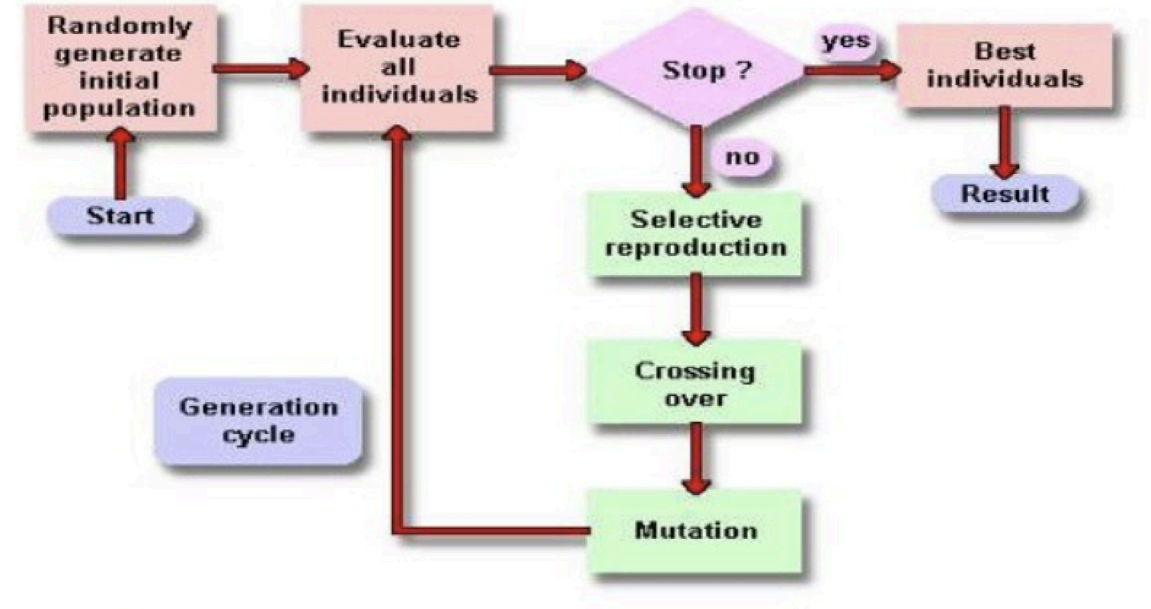
Attack Design

Threat model

We assume the attacker has black-box access to the target model i.e. the attacker is not aware of the model architecture, parameters, or training data, and is only capable of querying the target model with supplied inputs and obtaining the output predictions and their confidence scores. In other words, it cannot access any model information except inputs and outputs. It can only collect prediction results by feeding inputs.

Algorithm

They aim to minimize the number of modified words between the original and adversarial examples, but only perform modifications which retain semantic similarity with the original and syntactic coherence. To achieve these goals, instead of relying on gradient-based optimization, they introduce an attack algorithm that exploits population-based gradient-free optimization via genetic algorithms.

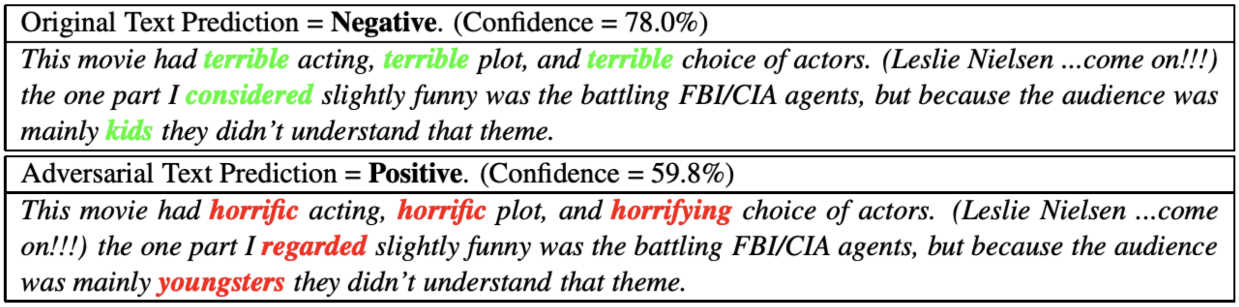


Genetic algorithms are inspired by the process of natural selection, iteratively evolving a population of candidate solutions towards better solutions.
The population of each iteration is called a generation. In each generation, the quality of population members is evaluated using a fitness function. “Fitter” solutions are more likely to be selected for breeding the next generation. The next generation is generated through a combination of crossover and mutation. Crossover is the process of taking more than one parent solution and producing a child solution from them; it is analogous to reproduction and biological crossover. Mutation is done in order to increase the diversity of population members and provide better exploration of the search space.

In order to select the best replacement word, the N nearest neighbors of the selected word are computed, according to the distance (Euclidean) in the embedding space. It is ensured that the neighbors are synonyms. Using another model, words that do not fit within the context are filtered out. From the remaining set of words, the one that will maximize the target label prediction probability when it replaces the word w is picked. The selection of which word to replace in the input sentence is done by random sampling with probabilities proportional to the number of neighbors each word has within Euclidean distance δ (threshold) in the embedding space.

Experimental Results

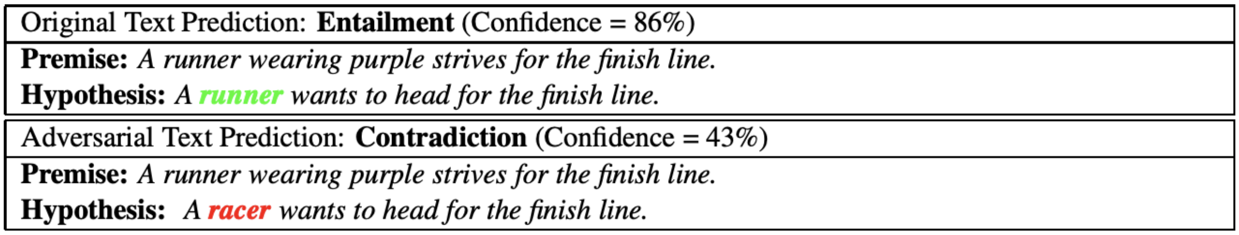
The authors trained models for the sentiment analysis (IMDB) and textual entailment (SNLI) classification tasks using the GloVe embedding vectors for training.



Original Text Prediction = **Negative**. (Confidence = 78.0%)
*This movie had **terrible** acting, **terrible** plot, and **terrible** choice of actors. (Leslie Nielsen ...come on!!!) the one part I **considered** slightly funny was the battling FBI/CIA agents, but because the audience was mainly **kids** they didn't understand that theme.*

Adversarial Text Prediction = **Positive**. (Confidence = 59.8%)
*This movie had **horrific** acting, **horrific** plot, and **horrifying** choice of actors. (Leslie Nielsen ...come on!!!) the one part I **regarded** slightly funny was the battling FBI/CIA agents, but because the audience was mainly **youngsters** they didn't understand that theme.*

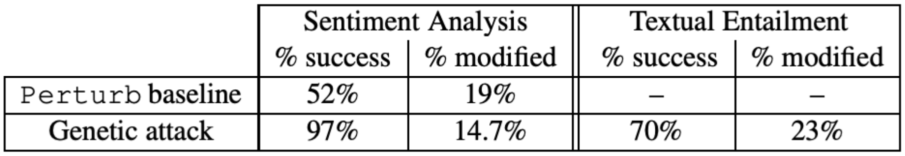
Table 1: Example of attack results for the sentiment analysis task. Modified words are highlighted in green and red for the original and adversarial texts, respectively.



Original Text Prediction: **Entailment** (Confidence = 86%)
Premise: *A runner wearing purple strives for the finish line.* **Hypothesis:** *A **runner** wants to head for the finish line.*

Adversarial Text Prediction: **Contradiction** (Confidence = 43%)
Premise: *A runner wearing purple strives for the finish line.* **Hypothesis:** *A **racer** wants to head for the finish line.*

Table 2: Example of attack results for the textual entailment task. Modified words are highlighted in green and red for the original and adversarial texts, respectively.



	Sentiment Analysis		Textual Entailment	
	% success	% modified	% success	% modified
Perturb baseline	52%	19%	–	–
Genetic attack	97%	14.7%	70%	23%

Table 3: Comparison between the attack success rate and mean percentage of modifications required by the genetic attack and perturb baseline for the two tasks.

They were able to achieve high success rate with a limited number of modifications on both tasks. In addition, the genetic algorithm significantly outperformed the Perturb baseline in both success rate and percentage of words modified, demonstrating the additional benefit yielded by using population-based optimization.

A human study is also conducted to validate these results.

It is demonstrated that despite the difficulties in generating imperceptible adversarial examples in the natural language domain, semantically and syntactically similar adversarial examples can be crafted using a black-box population-based optimization algorithm, yielding success on both the sentiment analysis and textual entailment tasks. Our human study validated that the generated examples were indeed adversarial and perceptibly quite similar.

This work also attempts to use adversarial training as a defense, but fails to yield improvement, demonstrating the strength and diversity of adversarial examples.These findings encourage researchers to pursue improving the robustness of DNNs in the natural language domain.

For more details, check out the original paper: https://arxiv.org/pdf/1804.07998.pdf

Related Work / Literature Survey

1) Explaining And Harnessing Adversarial Examples

Ian J. Goodfellow, Jonathon Shlens & Christian Szegedy Google Inc., Mountain View, CA

This paper addressed the vulnerability of neural networks and the effect of adversarial attacks on them. It demonstrated how networks were susceptible to small perturbation (noise) by which they could be fooled and lead to misclassify the model output with a high confidence. This was also generalised across different architectures and training sets. The paper went onto explain that neural networks’ vulnerability to adversarial perturbation is their linear nature internally. Generic regularization strategies such as dropout, pretraining, and model averaging do not confer a significant reduction in a model’s vulnerability to adversarial examples, but changing to nonlinear model families such as RBF networks can do so.
It also showed a simple and fast way of generating adversarial attacks.

https://arxiv.org/pdf/1412.6572.pdf

2) GenAttack: Practical Black-box Attacks with Gradient-Free Optimization

Moustafa Alzantot, Yash Sharma, Supriyo Chakraborty, Huan Zhang, Cho-Jui Hsieh, Mani Srivastava

This paper introduces GenAttack, a gradient-free optimization technique that uses genetic algorithms for synthesizing adversarial examples in the black-box setting. Though this paper targets the image domain, it is closely related with this paper as it follows the same algorithm, in the text domain. It explains the Theory of computation, Evolutionary algorithms as well as Computing methodologies to generate adversarial attacks.

https://arxiv.org/pdf/1805.11090.pdf

3) Towards Deep Learning Models Resistant to Adversarial Attacks
   Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, Adrian Vladu

This paper shows that the existence of adversarial attacks may be an inherent weakness of deep learning models. To address this problem, they study the adversarial robustness of neural networks through the lens of robust optimization. These methods let us train networks with significantly improved resistance to a wide range of adversarial attacks. They also suggest the notion of security against a first-order adversary as a natural and broad security guarantee.

https://arxiv.org/pdf/1706.06083.pdf

4) Practical Black-Box Attacks against Machine Learning
   Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, Ananthram Swami

This paper introduces the first practical demonstration of an attacker controlling a remotely hosted Deep Neural Network with no internal knowledge of the network. The only capability of their black-box adversary is to observe labels given by the DNN to chosen inputs. The attack strategy consists of training a local model to substitute for the target DNN, using inputs synthetically generated by an adversary and labeled by the target DNN. It uses the local substitute to craft adversarial examples, and find that they are misclassified by the targeted DNN with a high rate. It also finds that this black-box attack strategy is capable of evading defense strategies previously found to make adversarial example crafting harder and can be used as a defence technique.

https://arxiv.org/pdf/1602.02697.pdf

5) ZOO: Zeroth Order Optimization Based Black-box Attacks to Deep Neural Networks without Training Substitute Models
   Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, Cho-Jui Hsieh

Similar to the setting of training substitute models, this paper proposes an effective black-box attack that also only has access to the input (images) and the output (confidence scores) of a targeted DNN. However, different from leveraging attack transferability from substitute models, it proposes zeroth order optimization (ZOO) based attacks to directly estimate the gradients of the targeted DNN for generating adversarial examples. They use zeroth order stochastic coordinate descent along with dimension reduction, hierarchical attack and importance sampling techniques to efficiently attack black-box models. By exploiting zeroth order optimization, improved attacks to the targeted DNN can be accomplished, sparing the need for training substitute models and avoiding the loss in attack transferability.

https://arxiv.org/pdf/1708.03999.pdf