HOTFLIP: WHITE-BOX ADVERSARIAL EXAMPLES FOR TEXT CLASSIFICATION

UNIVERSITY OF OREGON

Javid Ebrahimi, Daniel Lowd, Dejing Dou
{Javid, lowd, dou}@cs.uoregon.edu

Anyi Rao
anyirao@smail.nju.edu.cn

Motivation

Adversarial examples are inputs designed to make a machine learning model perform poorly, and are often constructed by manipulating real-world examples. How can we manipulate discrete text representation to create adversarial examples? We focus on manipulating characters of text, by introducing differentiable string-edit operations, namely, *flip*, *insert*, and *delete*.

Examples of attacking a character-level neural text classifier:

South Africa's historic Soweto township marks its 100th birthday on Tuesday in a mood of optimism. 57% **World**
South Africa's historic Soweto township marks its 100th birthday on Tuesday in a moo**P** of optimism. 95% **Sci/Tech**
Chancellor Gordon Brown has sought to quell speculation over who should run the Labour Party and turned the attack on the opposition Conservatives. 75% **World**
Chancellor Gordon Brown has sought to quell speculation over who should run the Labour Party and turned the attack on the o**B**position Conservatives. 94% **Business**

HotFlip

Given an alphabet size of $|V|$, imagine the adversary is allowed to flip r characters in an input text with length L. Using a brute-force search, it would need to do $O(\frac{L!}{r!(L-r)!}|V|^r)$ forward passes to exhaust the search space and trick the classifier.

A Gradient-Based Surrogate Method:

Each change can be represented by a vector; for example, a character flip in the jth character of the ith word (a -> b) can be represented by this vector:

$$\vec{v}_{ijb} = (\vec{0},..;(\vec{0},..(0,..-1,0,..,1,0)_j,..\vec{0})_i;\vec{0},..)$$

where -1 and 1 are in the corresponding positions for the ath and bth characters of the alphabet, respectively. A first-order approximation of change in loss can be obtained from a directional derivative along this vector:

$$\nabla_{\vec{v}_{ijb}} J(\mathbf{x},\mathbf{y}) = \nabla_x J(\mathbf{x},\mathbf{y})^T \cdot \vec{v}_{ijb} = \frac{\partial J}{\partial x_{ij}}^{(b)} - \frac{\partial J}{\partial x_{ij}}^{(a)}$$

Deletes and inserts can be treated as a sequence of character flips, (e.g., an insert can be represented by a character flip, followed by more flips as characters are shifted to the right until the end of the word.)

Multiple Changes:

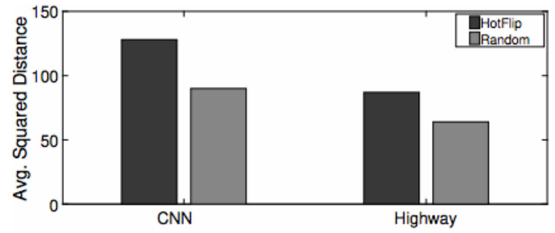
For additional changes we can perform one-shot, greedy, or beam search methods. For the beam search approach, our proposed adversary requires only O(br) forward passes and an equal number of backward passes, r being the budget and b being the beam width. In contrast, a naive loss-based approach requires computing the exact loss for every possible change at every stage of the beam search, leading to O(brL|V|) queries.

How Good Are the Gradients?

Gradients give a good estimate of the worst-case perturbations. The gradient-based approach needs an average of 1 more character flip to trick the classifier, but performs significantly faster.

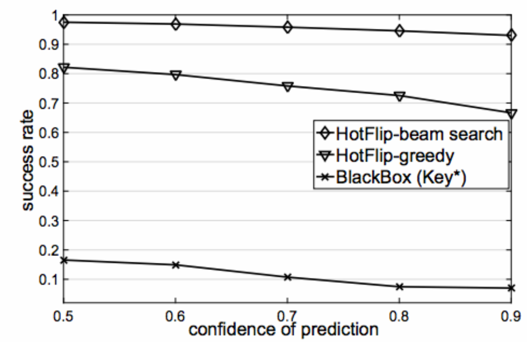
No. change(s)		1	2	3+
Loss-based	Time (s)	10.3	70.2	705
	Proportion	59%	29%	12%
Gradient-based	Time (s)	**0.11**	**0.93**	**2.7**
	Proportion	34%	29%	37%

Comparing the HotFlip direction and a random direction based on the average squared distance between the embedding of the original word, and the embedding of the modified word, found from the outputs of the CNN and highway layers, in the CharCNN-LSTM Architecture (Kim et al., 2016)



Experiments?

Experiments on AG's news corpus, on a neural classifier which achieves close to state-of-the-art result. Adversary's success rate for text classification can be measured by the misclassification rate of the classifier on the examples it had originally correctly classified.



Performing white-box adversarial training, we can make the model more robust, and even perform better on clean test data.

Method	Misc. error	Success rate
Baseline	8.27%	98.16%
Adv-tr (Miyato et al., 2017)	8.03%	87.43%
Adv-tr (black-box)	8.60%	95.63%
Adv-tr (white-box)	**7.65%**	**69.32%**

The adversary that we use at test time, which uses beam search, is strictly stronger than our model's internal adversary which uses a one-shot strategy; hence the success rate is still high. Adversarial training on real adversarial examples generated by HotFlip, is more effective than training on pseudo-adversarial examples created by adding noise to the embeddings (Miyato et al., 2017).

Human Perception

Our human evaluation experiment shows that character-based adversarial examples are much more likely to preserve the meaning of text than alter it. Concretely, the median accuracy of our participants for our text classification experiment decreased by only 1.78%, from 87.49% on clean examples to 85.71% on adversarial examples.

Embeddings Under Adversarial Noise

We can observe the impact of adversarial perturbation on word representation by inspecting nearest neighbor words (based on cosine similarity). A single adversarial change in the word often results in a big change in the embedding, which would make the word more similar to other words, rather than to the original word.

past → pas!t	Alps → llps	talk → taln	local → loral	you → yoTu
pasturing	lips	tall	moral	Tutu
pasture	laps	tale	Moral	Hutu
pastor	legs	tales	coral	Turku
Task	slips	talent	morals	Futurum

Word-Level Classification

HotFlip can naturally be adapted to attack word-level classifiers; given the need for semantic-preserving constraints, the adversary fails in most cases.

it's frustrating to see these guys who are obviously pretty clever waste their talent on parodies of things they probably thought were funniest when they were high. 83% **Negative Sentiment**
it's frustrating to see these guys who are obviously pretty **deft** waste their talent on parodies of things they probably thought were funniest when they were high. 65% **Positive Sentiment**

Machine Translation

In our follow-up work (Ebrahimi et al., 2018), we applied HotFlip to machine translation, and explored scenarios for targeted attacks.

src	In den letzten Jahren hat sie sich zu einer sichtbaren Feministin entwickelt.
adv	In den letzten Jahren hat sie sich zu einer sichtbaren **FbeminisMin** entwickelt.
src-output	In the last few years, they've evolved to a safe **feminist**.
adv-output	In the last few years, they've evolved to a safe **ruin**.
src	Ein Krieg ist nicht länger ein Wettbewerb zwischen Staaten, so wie es früher war.
adv	Ein Krieg ist nicht länger ein **erkBkaSzeKLlWmrt** zwischen Staaten, so wie es früher war.
src-output	A war is no longer a **competition** between states, like it used to be.
adv-output	A war is no longer a **throwaway** planet between states, as it used to be.

The adversary picks a word in the translation, and manipulates the input to generate a target word.

Acknowledgement:

This work was funded by ARO grant W911NF- 15-1-0265.

References:

- Javid Ebrahimi, Daniel Lowd, and Dejing Dou. 2018. On Adversarial Examples for Character-Level Neural Machine Translation. COLING
- Yoon Kim, Yacine Jernite, David Sontag, and Alexander M. Rush. 2016. Character-Aware Neural Language Models. AAAI
- Takeru Miyato, Andrew M Dai, and Ian Goodfellow. 2017. Adversarial training methods for semi-supervised text classification. ICLR