HotFlip: White-Box Adversarial Examples for Text Classification

1. **What is the primary lesson(s) you took away from this paper (avoid abstract level summary)?**

   The authors in this paper used a gradient-based optimization method (white-box attack)[1] to manipulate discrete text structure at its one-hot representation to generate adversarial examples to trick a character-level neural classifier and call it "HotFlip". The authors' method relies on an atomic flip operation, which swaps one token for another based on the one-hot input vectors' gradients. The authors, in their experiments, used two free datasets: AG's corpus of news articles and the Stanford Sentiment Treebank (SST) dataset, and used two free access classifiers: CharCNN-LSTM character-based classifier and CNN sentiment classifier for sentence classification. They found that only a few manipulations are needed to decrease the models' accuracy significantly and used at least three crowd workers in Amazon Mechanical Turk to show that their character-based adversarial examples rarely alter the meaning of a sentence.

   The authors also performed adversarial training to make their models more robust to attacks at test time and on clean test data and found that this adversarial training does make their models more robust to white-box attacks with the lowest misclassification error of 7.65% and the lowest success rate of adversary attack of 69.32%, among three models they implemented in the paper.

   The authors, lastly, demonstrated that their white-box attack, HotFlip, can be adapted to attack a word-level classifier using a few semantics-preserving constraints, where they were able to create only 41 examples (2% of the correctly-classified instances of the SST test set) with one or two flips.

2. **Could I have done this work if I had the idea why or why not?**

   I do not think I could have done this work if I had the idea, given my limited knowledge of CNN models (I am taking CS570 Deep Learning class this semester and CNNs on the syllabus).

3. **Is there anything I could do to repeat or validate? (Make notes of materials available from the paper- data sets, source code).**

   Yes, the authors are open-sourcing their attack and publicly published it on GitHub here: https://github.com/AnyiRao/WordAdver. The authors also used two free access datasets: AG's corpus of news articles (http://groups.di.unipi.it/~gulli/AG_corpus_of_news_articles.html) and the Stanford Sentiment Treebank (SST) dataset (https://nlp.stanford.edu/sentiment/index.html). They also used two open-source models on GitHub: CharCNN-LSTM character-based classifier: https://github.com/yoonkim/lstm-char-cnn and CNN sentiment architecture for sentence classification: https://github.com/yoonkim/CNN_sentence.

4. **What is my best idea for follow on work that I could personally do?**

   My best idea for the follow-on work that I could personally do is to first repeat the same work using their English datasets to develop a better understanding of how the attack works. Secondly, I could repeat the same work or part of it (sentiment analysis) using Arabic sentiment analysis datasets, examine the generated results, and test if this kind of adversarial text attack works in different languages like Arabic.

5. **What is my best idea for follow on work that I'd like to see the authors or others do?**

   My best idea for the follow-on work that I would like to see the authors or others do is to profoundly investigate the effectiveness of using adversarial training, meaning using the generated

[1] In white-box attacks, the attackers know the model architecture, parameters, or training datasets.

adversarial examples to train models to be more robust against these attacks. In Alzantot et al. (2018) paper, the authors found that adversarial training does not boost the models' robustness by implementing a black-box attack in the embedding space using the word-level replacements technique. Yet, in Ebrahimi et al. (2018) paper, the authors found that adversarial training does boost the models' robustness by implementing a white-box attack in the input space using the word-level replacements and character-level replacements techniques. The question that needs to be answered clearly does the attack type (black-box or white-box) plays a significant role in the effectiveness of adversarial training.

6. Any logistical experimental lessons I learned (this is a little different than #1 here you are looking for techniques)?

I liked how the authors took advantage of the open-source models/classifiers and datasets instead of starting from scratch (CharCNN-LSTM model, CNNs model, AG's dataset, and SST dataset).

I also liked the idea of having a human study as a validation method for their attack's results, and at the same time, they generated new results that could be a contribution by comparing them to the attack's results. The authors used at least three crowd workers in Amazon Mechanical Turk to show that their character-based adversarial examples rarely alter the meaning of a sentence.

7. How does this compare to the other papers we read? Most similar? How different? Other comparisons?

This paper, Ebrahimi et al. (2018), is the first paper I read that discusses the white-box attacks (the attackers are aware of the model architecture, parameters, or training datasets). Ebrahimi et al. (2018) is similar to Gao et al. (2018) in terms of the input space and the used method of character-level replacements and similar to Alzantot et al. (2018) in terms of the used method of word-level replacements as well. On the other hand, Ebrahimi et al. (2018) differs from both papers in terms of the attack type, where it implements a white-box attack and both papers implement black-box attacks.

8. What is your biggest criticism of the paper?

There is no big criticism of the paper. Yet, I would appreciate including more adversarial examples in the paper. The authors, for example, only included two adversarial examples of their character-based adversarial attack, which is their main contribution in this paper.

9. List 3 cited references or terms/concepts that you would be most interested in reading/learning more about.

I am interested in reading this reference (Goodfellow et al., 2015) because, so far, all the text adversarial attacks papers have cited this reference as a primary reference, if not the first in this field:
Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. ***Explaining and harnessing adversarial examples.*** *In Proceedings of ICLR.*

I am also interested in reading this reference (Belinkov and Bisk, 2018). The authors of this reference showed that character-level machine translation systems (MTs) are sensitive to random character manipulations, such as keyboard typos. It would be interesting to read this reference since the Arabic language and other languages are commonly translated from the English language:
Yonatan Belinkov and Yonatan Bisk. 2018. ***Synthetic and natural noise both break neural machine translation.*** *In Proceedings of ICLR.*

I would like to learn about distances/norms that are commonly used in machine learning, like the L_0 norm, L_1 (Manhattan Distance or Taxicab norm), L_2 (Euclidean Distance/norm), and L_∞ norm.