TEXTBUGGER: Generating Adversarial Text Against Real-world Applications

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

   The authors in this paper proposed TEXTBUGGER, a framework that could effectively and efficiently generate adversarial texts under white-box and black-box settings. In the white-box setting, the authors first looked for the important words by calculating the Jacobian matrix of the classifier and then selected an optimal perturbation from the generated five kinds of perturbations/attacks. While in the black-box setting, the authors first looked for the important sentences and then used a scoring function to identify important words to manipulate. The five general perturbations/attacks that the TEXTBUGGER framework generates are: inserting a space in a word, deleting a random character of a word, swapping two random adjacent letters in a word, replacing characters with visually similar characters or adjacent characters in the keyboard, and replacing a word with its top-*k* nearest neighbors in a context-aware word vector space.

   The authors evaluated the TEXTBUGGER framework on a few state-of-the-art machine learning models and popular real-world online applications, including sentiment analysis and toxic content detection. The experimental results showed that the TEXTBUGGER framework is very effective and efficient. For example, when targeting the Amazon AWS and Microsoft Azure platforms under black-box settings, the TEXTBUGGER framework achieved a 100% attack success rate on the IMDB movie reviews dataset. The authors also conducted a user study on their generated adversarial texts and showed that the TEXTBUGGER framework has little impact on human understanding. They further discussed the two potential defense strategies (spelling check and adversarial training) to defend against the above attacks, along with preliminary evaluations. Interestingly, the TEXTBUGGER still achieves a higher success rate even though many generated adversarial texts can be detected by spell-checking on multiple online platforms after correcting the misspelled words.

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 various model architectures, such as CNNs, bidirectional-LSTMs, and Regressions (I am taking CS570 Deep Learning class this semester).

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

   No, the authors did not open-source their attacks or defenses. Yet, there are a couple of replication attempts, like the TextAttack Python package that implements the attacks of TEXTBUGGER (https://textattack.readthedocs.io/en/latest/apidoc/textattack.attack_recipes.html?highlight=textbugger#textbugge) and a course project for a graduate student called 'LiKev12' at GitHub (https://github.com/LiKev12/CSE544T-Project-TextBugger). The authors used 3 publicly available datasets: the IMDB Movie Reviews dataset (https://ai.stanford.edu/~amaas/data/sentiment/), the Rotten Tomatoes Movie Reviews dataset (https://www.cs.cornell.edu/people/pabo/movie-review-data/), and lastly the Kaggle Toxic Comment Classification competition dataset (https://www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge/data).

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

   I do not have an idea for the follow-on work that I could personally do because the paper of Li et al. (2019) was a very comprehensive paper that covers all aspects of adversarial text attacks, covering Black and White attacks (word-level and character-level replacements/modifications) along with their proposed white and black defense mechanisms (spelling check and adversarial training).

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 improve the adversarial attacks/perturbations using "*a more sophisticated algorithm that takes advantage of language processing technologies, such as Syntactic Parsing, Named Entity Recognition, and Paraphrasing*" – the authors.

Another idea is to extend "*the existing attack procedure of finding and modifying salient words ... to beam search and phrase-level modification*" – the authors. It would be interesting to measure the effectiveness and the efficiency of the TEXTBUGGER attacks and defenses after such improvements and modifications.

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 provided the research community with an end-to-end study of the adversarial attacks in terms of white and black attacks (word-level and character-level replacements/modifications) and white and black defenses (spelling check and adversarial training). I also like how well the authors organized the paper, especially the grid-figures (sub-plots).

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

This paper by Li et al. (2019) is a very comprehensive paper that covers all aspects of adversarial text attacks, covering Black and White attacks along with their proposed defense mechanisms. This paper is similar to almost all the papers I have read in this course since most papers discuss attacks and defenses and share the same methods used and the input space with them. However, this paper differs from a few papers that target the embedding space instead of the input space, like Jia et al. (2019), Kuleshov et al. (2018), and Alzantot et al. (2018).

8. What is your biggest criticism of the paper?

There is no big criticism of the paper. Yet, the authors used a few acronyms that they have not defined and are unknown to readers, such as LR in models (Are they Linear or Logistic Regression models?) and CDF, *y*-axes in Figures 7, 10, and 14, are not defined (Are they 'Cumulative Distribution Functions' or something else?)

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

I like this paper very much because it covers all basics of adversarial attacks, and I am interested in reading these listed references to apprehend the proposed white/black attacks and defenses fully:

*Z. Gong, W. Wang, B. Li, D. Song, and W.-S. Ku, "**Adversarial texts with gradient methods**," arXiv preprint arXiv:1801.07175, 2018.*

*Y. Zhang and B. Wallace, **A sensitivity analysis of (and practitioners guide to) convolutional neural networks for sentence classification**, in IJCNLP, vol. 1, 2017, pp. 253–263.*

*N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami, **Practical black-box attacks against machine learning**, in Asia CCS. ACM, 2017, pp. 506–519.*

*Z. Zhao, D. Dua, and S. Singh, **Generating natural adversarial examples**, in ICLR, 2018.*

*D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar et al., **Universal sentence encoder**, arXiv preprint arXiv:1803.11175, 2018.*