Generating Natural Language Adversarial Examples

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

   The authors in this paper used a black-box population-based optimization algorithm to generate adversarial examples, preserving the original sentence semantics and syntactic via word replacements, that fool well-trained sentiment analysis and textual entailment models with success rates of 97% and 70%, respectively. The authors designed black-box attacks, meaning the attackers are unaware of the model architecture, parameters, or training datasets.

   The authors also introduced a population-based gradient-free optimization via genetic algorithms to optimize their attack, used GloVe to compute the nearest neighbors of the selected word according to the distance in the GloVe embedding space, used the Google one billion words language model to filter out words that do not fit within the context surrounding the word, picked the one word that will maximize the target label prediction probability when it replaces the word, and lastly, inserted the chosen word in place of the selected word in the sentence.

   The authors performed three experiments to measure the effectiveness of their generated adversarial examples: 1) They trained a sentiment analysis model (LSTM) on the IMDB reviews dataset. 2) They trained the textual entailment model (SNLI) on Stanford Natural Language Inference (SNLI) dataset. 3) They adversarially trained LSTM model by adding the adversarial examples to the training dataset to increase the robustness of the model; this failed to add robustness to the model.

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

   I think if I had the idea first, I could have done this work. The high-level idea of the authors' work in this paper is to replace specific words in the original sentence with very close ones using GloVe's most similar words of a selected word to be replaced, and I have good experience in working with the GloVe model and its features. Plus, both authors' experiments were based on baseline models and datasets that are publicly available. They only developed their attack algorithm.

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 *"to encourage research in training DNNs robust to adversarial attacks in the natural language domain."* – the authors. They publicly published their attack on GitHub here: https://github.com/nesl/nlp_adversarial_examples.

   The authors also used two free datasets: IMDB and Stanford Natural Language Inference (SNLI) dataset, and used two publicly available baseline models: LSTM (https://github.com/sonyisme/keras-recommendation/blob/master/keras-master/examples/imdb_lstm.py) and SNLI baseline model (https://github.com/Smerity/keras_snli/blob/master/snli_rnn.py).

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 adversarial training, which uses the generated adversarial examples as a part of the training dataset. The authors did a study where they added to their training dataset 1000 generated adversarial examples. They then trained their sentiment analysis model from scratch on the new training dataset, but they found that this adversarial training did not improve the robustness of the model.

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 utilized the textual entailment NLP task in their experiments, where they trained a textual entailment model using the Stanford Natural Language Inference (SNLI) corpus.

I also liked the idea of having a user 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.

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

This is my first paper on my reading pool, but in future writeups, I will compare this paper to others we read in terms of similarity, difference, or other criteria.

8. **What is your biggest criticism of the paper?**

There is no big criticism of the paper. Still, I would appreciate a short context of the optimization algorithms after mentioning them for the first time in the manuscript, such as gradient-based optimization and gradient-free optimization algorithms. I had to go and look each one up to understand the authors' work.

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 (Mrksic et al., 2016). The authors used their counter-fitting method to post-process the adversary's GloVe vectors to ensure that the nearest neighbor words are synonyms:
Nikola Mrksic, Diarmuid O Seaghdha, Blaise Thomson, Milica Gasic, Lina Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. ***Counter-fitting word vectors to linguistic constraints.*** *In North American Chapter of the Association for Computational Linguistics.*

I am also interested in reading this reference (Chelba et al., 2013). The authors used Google's one billion words language model to filter out words that do not fit within the context surrounding the word by ranking the candidate words based on their language model scores when they fit within the replacement context.
Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. 2013. ***One billion word benchmark for measuring progress in statistical language modeling.*** *arXiv preprint arXiv:1312.3005.*

I would like to learn about optimization algorithms. The authors in this paper discussed the cons and pros of using gradient-based optimization vs. gradient-free optimization via genetic algorithms.