Is Bert Really Robust? A Strong Baseline for Natural Language Attack on Text Classification and Entailment

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

The authors in this paper proposed a black-box, simple but strong baseline, TextFooler, to efficiently generate high-profile utility-preserving adversarial examples that effectively force the targeted deep learning models to make wrong predictions. The proposed approach for this baseline, TextFooler, is to generate adversarial text generation efficiently in two main steps: identifying and ranking important words and introducing word transformation that preserves similar semantic meaning with the original word, fits within the surrounding context, makes the target model make wrong predictions.

The authors also evaluated the TextFooler on three state-of-the-art deep learning classifiers over seven datasets: five popular Text Classification tasks and two Textual Entailment tasks. It achieved a state-of-the-art attack success rate and perturbation rate. These three deep learning classifiers are word-based Convolutional Neural Networks (word-CNN), word-based Long Short-Term Memory (word-LSTM), and Bidirectional Encoder Representations from Transformers (BERT). Generally, TextFooler can always reduce the accuracy from these state-of-the-art models to below 15% with less than a 20% word perturbation ratio.

The authors, lastly, proposed four comprehensive automatic evaluations and three human evaluations of the language of adversarial attacks to evaluate their system's effectiveness, efficiency, and utility-preserving properties. These four automatic evaluations include after-attack accuracy, percentage of perturbed words, semantic similarity, and query number. The three human evaluations of the language of adversarial attacks are judging the grammaticality score, assigning classification labels, and judging the similarity score between examples. The overall agreement between the labels of the original and the adversarial sentence is relatively high, 92% on RTMR and 85% on SNLI.

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 LSTMs models (word-based LSTMs), CNNs models (word-based CNNs), and BERT model (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).**

Yes, the authors have open-sourced their attacks (TextFooler) and published them on GitHub here: https://github.com/jind11/TextFooler. The authors in this work used **seven** publicly available datasets (**five** datasets for Text Classification task and **two** datasets for Textual Entailment task): AG's News Corpus [1], Fake News Detection (Fake) dataset [2], Rotten Tomatoes Movie Review (RTMR) dataset [3], IMDB Movie Review dataset [4], Yelp Polarity (Yelp) dataset [5], the Stanford Natural Language Inference (SNLI) dataset [6], and the Multi-Genre NLI (MultiNLI) dataset [7].

[1] AG's News: http://groups.di.unipi.it/~gulli/AG_corpus_of_news_articles.html.
[2] Fake News: https://www.kaggle.com/c/fake-news/data.
[3] RTMR: https://www.cs.cornell.edu/people/pabo/movie-review-data/.
[4] IMDB: https://datasets.imdbws.com/.
[5] Yelp: https://www.tensorflow.org/datasets/catalog/yelp_polarity_reviews.
[6] SNLI: https://nlp.stanford.edu/projects/snli/.
[7] MultiNLI: https://cims.nyu.edu/~sbowman/multinli/.

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 (text classification) using Arabic text classification datasets, examine the generated results, and test if this kind of adversarial text attack works in different languages like the Arabic language.

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

I do not have an idea for the follow-on work I would like to see the authors or others do.

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 explained everything and provided an end-to-end study. This paper by Jin et al. (2020) is one of the very well-written papers I have read in this course. I also like how the authors cited the original authors and their papers for every technique/method/setting they used.

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

This paper by Jin et al. (2020) is the second paper I read that utilizes the BERT language model to generate adversarial attacks in a black-box setting. The first paper was by Garg & Ramakrishnan (2020), who used TextFooler as their baseline. These two papers are very similar regarding the targeted space (input), the attack type (black-box), and the method used (word-level replacements). Yet, they are completely different in the used approaches. Garg & Ramakrishnan (2020) used four attacks: replacing a word, inserting a word, replacing or inserting a word, and replacing and inserting a word, whereas Jin et al. (2020) only substituted a selected word with another candidate word.

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

There is no big criticism of the paper. Yet, Table 9 (the qualitative comparison of adversarial attacks with and without the semantic similarity constraint) is unclear because both rows showed only adversarial attacks with semantic similarity constraints. Maybe they meant that the upper or the lower row shows the adversarial attacks without semantic similarity constraint.

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

I am interested in reading these three references listed below because they are relevant and have been cited in this paper, and I do not have them on my to-read list.

Chen, Q.; Zhu, X.; Ling, Z.; Wei, S.; Jiang, H.; and Inkpen, D. 2016. ***Enhanced LSTM for natural language inference.*** *arXiv preprint arXiv:1609.06038.*

Conneau, A.; Kiela, D.; Schwenk, H.; Barrault, L.; and Bordes, A. 2017. ***Supervised learning of universal sentence representations from natural language inference data****. arXiv preprint arXiv:1705.02364.*

Niven, T., and Kao, H.-Y. 2019. ***Probing neural network comprehension of natural language arguments****. arXiv preprint arXiv:1907.07355.*

I would like to learn more about the BERT model, especially how to fine-tune it on custom corpora. Another thing I would like to learn is the Masked Language Models (MLMs) tasks and how to use such tasks to produce presentative, textual word embeddings for a given corpus.