Toward Mitigating Adversarial Texts

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

The authors in this paper proposed a defense mechanism against back-box adversarial attacks/examples in English using a novel approach of spell-checking systems that utilize frequency and contextual information for correcting nonword misspellings. The proposed approach combines Levenshtein distance (edit distance), frequency counts (unigrams and bigrams), and contextual similarity techniques for correcting misspellings. The proposed method first suggests candidates for misspelled nonwords using a dictionary with an edit distance up to a given threshold. Then, the approach ranks all the candidates using a scoring function. The scoring function computes the average of the Candidate Frequency Score (CF), Preceding Bigram Frequency Score (PB), Succeeding Bigram Frequency Score (SB), and Contextual Similarity Score (CS). Lastly, the candidate with the highest score is chosen as the correct spelling.

The authors evaluated their proposed defense mechanism for black-box attacks on the two publicly available datasets: Yelp Reviews Polarity and the Yelp Reviews Full datasets, using adversarial examples generated by various black-box attacks such as Gao et al. (2018) and Belinkov and Bisk (2018). The authors also used two models introduced by Gao et al. (2018) and Zhang et al. (2015): the Word-level Bi-LSTM model and the Character-level CNN model, respectively. Finally, the authors used Word2Vec with a Continuous Bag-of-Words (CBOW) model that is trained on the used corpora of Yelp reviews datasets to produce the Word Embeddings and used the extracted dictionary of tokens as a dictionary for the misspelling corrections.

The authors' proposed defense mechanism increased the classification accuracy of the sentiment analysis task by an average of 26.56% on the Yelp Reviews Polarity dataset and 16.27% on the Yelp Reviews Full dataset. The models' accuracies had dropped when the attacks of Gao et al. (2018) and Belinkov and Bisk (2018) were applied from 93.75% to 62.10% on the Yelp Reviews Polarity dataset and 60.93% to 40.23% on the Yelp Reviews Full dataset, respectively. Additionally, the proposed defense approach outperformed six publicly available, state-of-the-art spelling correction tools: Aspell, Python Auto-corrector, Hunspell, CSpell, TextBlob, and Google's spell-checker, by at least 25.56% in average correction accuracy when these six spelling correction tools were used instead of the proposed defense method.

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 or part of it. The high-level idea of the authors' work in this paper is to defend models against character-level black-box adversarial attacks using spell-checking systems. The authors' proposed approach combines Levenshtein distance (edit distance), frequency counts, and contextual similarity techniques (Word2Vec with CBOW) for correcting misspellings. The authors' experiments were based on two models (Gao et al., 2018: the Word-level Bi-LSTM model and Belinkov and Bisk, 2018: the Character-level CNN model) and two datasets (Yelp Reviews Polarity and the Yelp Reviews Full) that are publicly available. They only developed their defense mechanism, which is explained above.

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 have not published their defense mechanism. Yet, Belinkov and Bisk's Character-level CNN model is here: https://github.com/boknilev/dsl-char-cnn, Gao et al.'s attacks are here: https://github.com/QData/deepWordBug, and Yelp reviews datasets are here: http://goo.gl/JyCnZq.

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 character-level attacks and defense mechanism work. Secondly, I could repeat the same work or part of it (sentiment analysis) using Arabic sentiment analysis datasets, examine the generated results, and test whether this kind of character-level attacks and defense mechanisms work in 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 measure the effectiveness of this defense mechanism using spell-checking systems to decrease the success of the character-level adversarial attacks on the models that have been trained adversarially, i.e., models that have been trained on adversarial examples to gain robustness. Will this defense mechanism be more efficient than the adversarial training for the natural language models? How effective is the spell-checking defense mechanism with adversarial training if the two approaches are combined?

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 used six publicly available spelling correction tools to benchmark their spell-checking defense mechanism. Their approach interestingly outperformed these state-of-the-art spelling correction tools.

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

This paper, Alshemali & Kalita (2019), is the first paper I read about defense mechanisms against adversarial attacks, especially back-box attacks where the attackers are unaware of the model architecture, parameters, or training datasets. This paper used primarily two papers: Gao et al. (2018) and Belinkov and Bisk (2018). I have already read Gao et al. (2018), which introduced black-box attacks in the input space using the method of character-level replacements. I have not read Belinkov and Bisk (2018) yet, but it is on my to-read list; this paper discussed character-based adversarial attacks for Neural Machine Translation (NMT) models.

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

There is no big criticism of the 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 (Hosseini et al., 2017), where the authors introduced spelling errors into the Perspective API that undermine toxic comment detection systems:

Hossein Hosseini, Sreeram Kannan, Baosen Zhang, and Radha Poovendran. ***Deceiving Google's perspective API built for detecting toxic comments.*** *In Proceedings of the Conference on Computer Vision and Pattern Recognition, 2017.*

I am also interested in reading this reference (Fivez et al., 2017), where the authors of this work proposed a context-sensitive spelling correction method for clinical text in English. Alshemali & Kalita (2019) followed this work to compute their Contextual Similarity Score (CS).

Pieter Fivez, Simon Suster, and Walter Daelemans. ***Unsupervised context-sensitive spelling correction of English and Dutch clinical free-text with word and character n-gram embeddings****. Biomedical Natural Language Processing Workshop, pages 143–148, 2017.*