Journal of the American Medical Informatics Association, 26(6), 2019, 577–579
doi: 10.1093/jamia/ocz013
Advance Access Publication Date: 11 April 2019
Correspondence

Correspondence

Comment on: "Deep learning for pharmacovigilance: recurrent neural network architectures for labeling adverse drug reactions in Twitter posts"

Arjun Magge,[1] Abeed Sarker,[2] Azadeh Nikfarjam,[2] and Graciela Gonzalez-Hernandez[2]

[1]College of Health Solutions, Arizona State University, Scottsdale, Arizona, USA, and [2]Perelman School of Medicine, University of Pennsylvania, Philadelphia, Pennsylvania, USA

Received 28 November 2018; Editorial Decision 20 December 2018; Accepted 21 January 2019

Dear Editor,

We read with great interest the article by Cocos et al.[1] In it, the authors use one of the datasets made public by our lab in parallel with a publication in *Journal of the American Medical Informatics Association*,[2] referred to by them as the Twitter ADR Dataset (v1.0) (henceforth the ADRMine Dataset). Cocos et al use state-of-the-art recurrent neural network (RNN) models for extracting adverse drug reaction (ADR) mentions in Twitter posts. We commend the authors for their clear description of the workings of neural models, and on their experiments on the use of fixed versus trainable embeddings, which can be very valuable to the natural language processing (NLP) research community. We believe that using deep learning models offer greater opportunities for mining ADR posts on social media.

However, there are key choices made by the authors that require clarification to avoid a misunderstanding on the impact of their findings. In a nutshell, because the authors did not use the ADRMine Dataset in its entirety, discarding upfront all tweets with no human annotations (ie, those that do not contain any ADRs), the resulting train and test sets are biased toward the positive class. Thus, the performance measures reported for the task in Cocos et al are not comparable to those reported in Nikfarjam et al,[2] contrary to what the manuscript reports.

After discarding tweets with no human annotation from the ADRMine Dataset, the authors downloaded available tweets from Twitter, and added a small set (203 tweets) to form the dataset used for their experiments. While downloading from Twitter results in an almost unavoidable reduction in the dataset size—as not all tweets are available as time goes by—it would not generally affect the class balance. The elimination of the tweets with no human annotations from the ADRMine Dataset, however, is a choice that is not discussed by Cocos et al, even though it severely impacts the positive-to-negative class balance of the dataset, leaving it at the 95 to 5 that they report, and, as our experiments show, has a significant impact on the reported performance. Our comparisons of ADRMine with the system proposed by Cocos et al reveal that, actually, when the two systems are employed on the dataset with the original balance, ADRMine[2] performs significantly better than their proposed approach (last two rows of Table 1). Thus, the claim in the Results and Conclusion sections of Cocos et al that their model "represents new state-of-the-art performance" and that "RNN models . . . establish new state-of-the-art performance by achieving statistically significant superior F-measure performance compared to the CRF-based model" is premature. We expand on these points next.

To give some context to the ADRMine dataset, it contains a set of tweets collected on medication name as a keyword. Retweets were removed, and tweets with a URL were omitted, given that our analysis showed that they were mostly advertisements. To balance the data in a way that reflected what was automatically possible at the time, a binary classifier with precision around 0.4-0.5 was assumed. Thus, negative (non-ADR) instances were kept at around 50%, down from approximately 89% non-ADR tweets that come naturally when collecting on medication name as a keyword,[2] a balance one would expect for this task utilizing state-of-the-art automatic methods for classification before attempting extraction. It is thus a realistic, justified, balance.

Regarding the Cocos et al approach, although controlled experiments training with different ratios of class examples are not unusual in machine learning, results for different positive-to-negative ratios are usually reported and are noted upfront. Cocos et al use a 95-to-5 positive-to-negative split, and only report on the performance on this altered dataset, making no mention of the alteration or class imbalance in the abstract. The statement in the abstract summarizes their results as follows: "Our best-performing RNN

Table 1. Performance comparison of NERs under different training and testing modes

Mode	Dataset Size	Precision	Recall	F_1-score
Cocos et al on *MostlyPos* dataset as published	844 tweets	0.70 (0.66-0.74)	0.82 (0.76-0.89)	0.75 (0.74-0.76)
October 2018: train $_{MostlyPos}$ and test $_{MostlyPos}$	526 tweets	0.76 (0.70-0.82)	0.72 (0.63-0.81)	0.73 (0.70-0.76)
October 2018: train $_{MostlyPos}$ and test $_{Standard}$	644 tweets	0.60 (0.54-0.65)	0.70 (0.62-0.77)	0.63 (0.60-0.66)
October 2018: train $_{Standard}$ and test $_{Standard}$	1012 tweets	0.73 (0.66-0.79)	0.60 (0.52-0.68)	0.64 (0.62-0.66)
Cocos et al on ADRMine Dataset	1784 tweets	0.68 (0.62-0.73)	0.69 (0.62-0.75)	0.67 (0.66-0.69)
ADRMine on ADRMine Dataset as published[1]	1784 tweets	0.76	0.68	0.72

Values are mean (95% confidence interval). Scores were achieved by each model over 10 training and evaluation rounds. *MostlyPos* refers to how the dataset is used by Cocos et al (ie, removing tweets without span annotations), hence leaving mostly positive tweets. *Standard* refers to the dataset including a roughly 50-50 balance of positive to negative tweets as in Nikfarjam et al,[2] and the balance of the *ADRMine Dataset*.

model ... achieved an approximate match F-measure of 0.755 for ADR identification on the dataset, compared to 0.631 for a baseline lexicon system and 0.65 for the state-of-the-art conditional random fields model." Although further in the manuscript Cocos et al refer to having implemented a CRF model "as described for previous state-of-the-art results," citing Nikfarjam et al,[2] the statement in the abstract could be misconstrued as directly comparing it to Nikfarjam et al, which is the state-of-the-art conditional random fields (CRF) model. In reality, the results are not comparable, given the changes to the dataset. Their implementation of a CRF model must have been significantly different to ADRMine as described in Nikfarjam et al, given that the reported performance in Cocos et al for a CRF model (0.65) is much lower than when both systems are used on the unaltered ADRMine Dataset, as our experiments show (last two rows of Table 1).[2] Please note that Cocos et al did not make available their CRF model implementation, so any differences to the ADRMine model could not be verified directly, only inferred from the reported results. The binaries of ADRMine were available at the time of publication, and we have since made available the full code to facilitate reproducibility.[a]

In machine learning research, authors decide how the model is trained and how the data are algorithmically filtered before training, apply accepted practices for balancing the data, or include additional weakly supervised examples.[3] However, such methods are applied to the training data only, leaving the evaluation data intact in order to be able to compare approaches. By excluding tweets that are negative for the presence of ADRs and other entities from their training, the authors built a model that is biased to the positive class. This might not be immediately obvious in Cocos et al, as the model is evaluated against a similarly biased test set. However, when run against the balanced test set, the problem becomes evident. The authors do note this, stating that "including a significant number of posts without ADRs in the training data produced relatively poor results for both the RNN and baseline models," but they did not include a report of these results or altered their experimental approach to make this more evident.

To illustrate the impact of the dataset modifications on the overall results, we ran the training and evaluation experiments on the ADRMine Dataset for tweets available as of October 2018 using the authors' publicly available implementation[b] and summarize the results in Table 1. Under the same settings as Cocos et al (eliminating virtually all tweets in the negative class), the performance reported (row 1) and our replication (row 2) can be considered a match with a slight drop that could be attributed to fewer tweets available as of October 2018 compared with when they ran it. However, evaluating the Cocos et al model on the balanced test set (row 3) shows a drop of 10 percentage points compared with evaluating against the mostly positive set (row 2). Training on all available positive and negative tweets from the October 2018 set (row 4) leads to an improved model but continues to show significantly lower performance (0.64) with respect to when the same model is trained and tested on the biased set (0.73 in row 2). Additionally, and to be able to do a direct comparison, we trained and tested the Cocos et al system as provided by them (except for the download script) on the original, balanced, ADRMine Dataset containing 1784 tweets. We found a mean performance of 0.67 over 10 runs (row 5), 5 points lower than the 0.72 F_1-score reported in Nikfarjam et al on the same dataset (row 6).[2]

Furthermore, referring to the ADRMine Dataset,[2] Cocos et al report, "Of the 957 identifiers in the original dataset...," which is incorrect. The original dataset, publicly available and unchanged since its first publication in 2015, contains a total of 1784 tweets (1340 in the training set and 444 in the evaluation or test set). As of October 2018, 1012 of the 1784 original training set tweets were still available in Twitter (including 267 of the 444 original evaluation tweets). Cocos et al do not mention the additional 827 tweets that were in the ADRMine Dataset, even though many of them were still available at the time of their publication. They used only 149 tweets from the 444 in the evaluation set. From our analysis, the 957 mentioned in Cocos et al correspond to the number of tweets in the ADRMine Dataset that are manually annotated for the presence of ADRs and other entities, such as indications, drug, and other (miscellaneous) entities. The rest (827 tweets) mentioning medications but with no other entities present, are discarded upfront, as can be observed by running Cocos et al's code, the download_tweets.py script. Although the Cocos et al code points researchers to the original site to download the ADRMine Dataset, once they move on to the said script with that data, they lose all the unannotated negative tweets. The authors do not discuss the rationale as to why the dataset was modified in such a manner. From the time that Cocos et al was published, subsequent papers have also used the 95-to-5 positive-to-negative split, presumably because they reuse the python script.[4–7] We have made available with this letter, a modification to the download_tweets.py script that will keep previously discarded tweets.[c]

In conclusion, the performance reported for the RNN model in Cocos et al is not comparable to any prior published approach, and

a https://github.com/azinik/ADRMine Accessed November 21, 2018.

b https://github.com/chop-dbhi/twitter-adr-blstm Accessed November 21, 2018.

c https://bitbucket.org/pennhlp/twitter-adr-blstm-download-tweets Accessed November 21, 2018.

in effect, when trained and tested with the full dataset, its performance (0.64) is significantly lower than the state of the art for the task (0.72).[2] ADR mentions are very rare events on social media, as has become evident through shared tasks on ADR detection in social media. Even after three years, the best classifier reaches only a precision of 0.44, recall of 0.63, for an F-measure of 0.52.[8] The upfront stripping of negative examples, whereby 95% of the dataset contains at least 1 ADR or indication mention, as done in Cocos et al, results in an extremely biased dataset, which in turn results in a model biased to the positive class that does not reflect any realistic deployment of a solution to the original problem.

FUNDING

This work was supported by National Institutes of Health National Library of Medicine grant number 5R01LM011176. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Library of Medicine or National Institutes of Health.

AUTHOR CONTRIBUTORS

AM first noted the data use problem, ran the experiments and wrote the initial draft of the manuscript. AS and AN contributed to some sections and made edits to the manuscript. GG designed the experiments and wrote the final version of the manuscript.

Conflict of interest statement: None declared.

REFERENCES

1. Cocos A, Fiks AG, Masino AJ. Deep learning for pharmacovigilance: recurrent neural network architectures for labeling adverse drug reactions in Twitter posts. J Am Med Inform Assoc 2017; 24 (4): 813–21.
2. Nikfarjam A, Sarker A, O'Connor K, Ginn R, Gonzalez G. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *J Am Med Inform Assoc* 2015; 22 (3): 671–81.
3. Galar M, Fernandez A, Barrenechea E, Bustince H, Herrera F. A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches. *IEEE Trans Syst Man Cybern Part C* 2012; 42 (4): 463–84.
4. Gupta S, Gupta M, Varma V, Pawar S, Ramrakhiyani N, Palshikar GK. Multi-task learning for extraction of adverse drug reaction mentions from tweets. In: *European Conference on Information Retrieval*, 2018; 59–71.
5. Shashank G, Sachin P, Nitin R, Girish Keshav P. Vasudeva V. Semi- supervised recurrent neural network for adverse drug reaction mention extraction. *BMC Bioinformatics* 2018; 19 (8): 212.
6. Gupta S, Gupta M, Varma V, Pawar S, Ramrakhiyani N, Palshikar GK. Co-training for extraction of adverse drug reaction mentions from tweets. In: *European Conference on Information Retrieval*, 2018; 556–62.
7. Shaika C, Chenwei Z, Yu PS. Multi-task pharmacovigilance mining from social media posts In: *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, 2018; 117–26.
8. Weissenbacher D, Sarker A, Paul MJ, Gonzalez-Hernandez G. Overview of the third social media mining for health (SMM4H) shared tasks at EMNLP 2018. In: *Proceedings of the 2018 EMNLP Workshop SMM4H: The 3rd Social Media Mining for Health Applications Workshop and Shared Task*, 2018; 13–16.