DeepADEMiner: A Deep Learning Pharmacovigilance Pipeline for Extraction and Normalization of Adverse Drug Effect Mentions on Twitter

Center for Health Language Processing

Arjun Magge[1], Elena Tutubalina[2], Zulfat Miftahutdinov[2], Ilseyar Alimova[2], Anne Dirkson[3], Suzan Verberne[3], Davy Weissenbacher[1], Graciela Gonzalez-Hernandez[1]

[1] Department of Biostatistics, Epidemiology and Informatics, University of Pennsylvania, Philadelphia, PA
[2] Kazan Federal University, Kazan, Russia
[3] LIACS, Leiden University, Leiden, Netherlands

Motivation and Challenge

- Research on pharmacovigilance from social media data has focused on mining adverse drug effects (ADEs) using classification and named entity recognition (NER) techniques in a pipeline architecture.
- This is an extremely challenging task because ADE mentions are rare due to other general domain posts, advertisements and ambiguity.
- The goal of detecting ADE signals for informing public policy has also been impeded largely by limited end-to-end solutions for large-scale analysis of social media reports for different drugs.

Objectives

- We evaluate the utility of including an ADE classifier as the first step of a pipeline to tackle the imbalance in the data.
- We demonstrate the impact of training the NER using varying ratios of ADE positive (hasADE) to ADE negative (NoADE) tweets on the end-to-end ADE extraction and normalization performance to measure the effect of tweet level class imbalance on NER performance.
- We establish state-of-the-art performance on an end-to-end ADE extraction and normalization pipeline. We make the end-to-end pipeline available to the public as an API endpoint and an online interactive tool. [1]

Materials

- We present a dataset for training and evaluation of ADE pipelines containing 29,284 tweets annotated with 2,265 ADE mentions where the ADE distribution is closer to the average `natural balance' with ADEs present in about 7% of the Tweets. The annotated ADE mentions also contain the corresponding normalized medical term in the MedDRA ontology. [2]
- The dataset is split into 18,300 (62.5%) tweets for training and 10,984 (37.5%) tweets for testing.

Methods

Supervised Training: The pipeline architecture consists of three individual layers of transformer models used to:
(1) filter posts that contain ADE
(2) extract spans of ADE mentions
(3) normalize ADE mentions to MedDRA preferred terms

Normalizer Training: To design the system to normalize ADE mentions that are not in the training dataset, we use a semi-supervised training procedure that includes terms from the MedDRA ontology and related terms integrated from UMLS vocabulary.

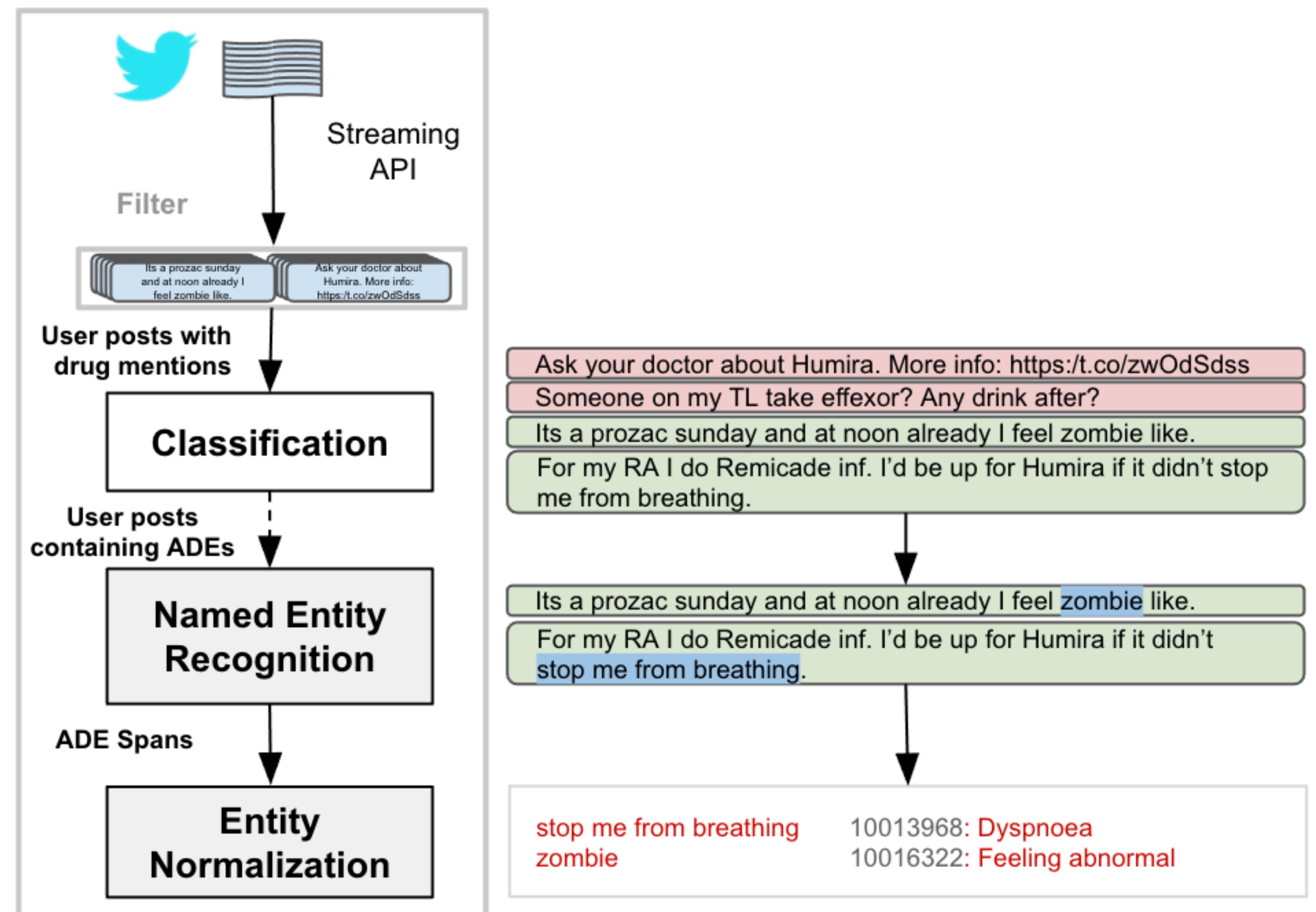

Fig 1. System architecture used for the ADE extraction pipeline

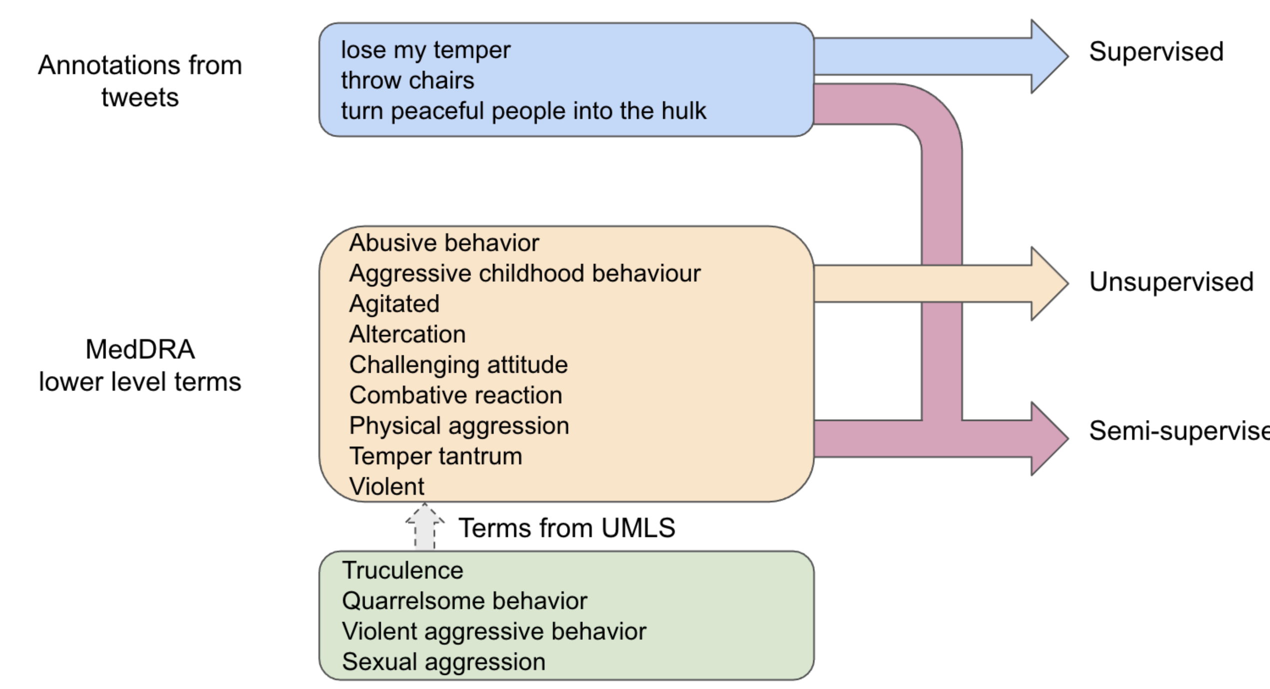

Fig 2. Normalization Architecture describing the three methods of training based on annotations from social media and terms from MedDRA and UMLS

Results and Conclusion

- The easiest way to obtain better performance across all components is to switch to transformer-based classifiers and sequence taggers. However, it comes at the cost of inference time.
- Experiments from the variation in proportion of tweets for the NER suggest that a ratio of 1 tweet with ADE to 2 tweets containing no ADEs result in optimal performance.
- Combined experiments of classifier and NER suggests that inclusion of the ADE tweet level classifier is beneficial to the overall pipeline.
- Inclusion of labels from MedDRA and UMLS was beneficial to improve normalization performance and overall performance.
- Our deep learning architecture achieves a classification performance of F_1=0.63, span extraction performance of F_1=0.44 and an end-to-end performance i.e., classification, extraction and normalization F_1=0.34.

Method	Precision	Recall	F_1-score
Glove	0.432	0.171	0.245
Twitter Health	0.571	0.182	0.276
FastText	0.741	0.192	0.304
BERT	**0.785**	**0.200**	**0.319**

Fig 3. ADE span extraction performance using overlapping precision, recall and F1-scores when trained on the full dataset in the absence of a classifier.

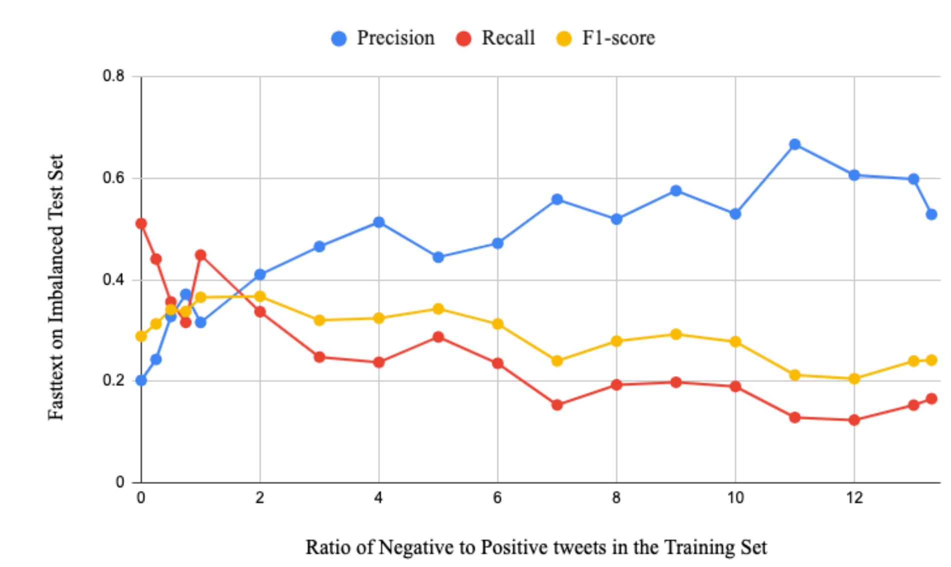

Fig 4. The chart shows how the variation in proportion of tweets in noADE and hasADE classes affects the performance of the ADE span extraction system.

Method	Configuration	Acc (overall)	Acc (train)	Acc (test)
FastText	Unsupervised	0.414	0.425	0.402
	Supervised	0.495	0.442	0
	Semi-supervised	0.521	0.551	0.411
BERT	Unsupervised	0.441	0.447	0.415
	Supervised	0.590	**0.653**	0
	Semi-supervised	**0.612**	0.638	**0.497**

Fig 5. Normalization task performance on the test set operating under the assumption where extracted spans are available.

Task	Corpus	SOTA ($P/R/F_1$)	DeepADEMiner ($P/R/F_1$)
Classification	SMM4H [3]	0.62 / 0.65 / 0.64	0.63 / 0.67 / 0.65
	HLP-ADE-v1	-	0.61 / 0.64 / 0.63
NER	SMM4H [3]	0.79 / 0.72 / 0.75	0.82 / 0.76 / 0.78
	HLP-ADE-v1	-	0.53 / 0.38 / 0.44
Resolution	SMM4H [3]	0.48 / 0.45 / 0.46	0.52 / 0.49 / 0.51
	HLP-ADE-v1	-	0.41 / 0.29 / 0.34

Fig 6. Performance comparison of the components introduced in this work with state-of-the-art (SOTA) implementations and datasets

Paper, Resources, References and Acknowledgements

[1] DeepADEMiner: Software, Demo and API available at https://healthlanguageprocessing.org/pubs/deepademiner/
[2] MedDRA ontology https://www.meddra.org/
[3] Klein et al. Proceedings of the Fifth Social Media Mining for Health Applications Workshop & Shared Task. ACL 2020

Research supported by National Institutes of Health (NIH). The content is solely the responsibility of the authors and does not necessarily represent the official views of NIH.