Journal of the American Medical Informatics Association, 28(10), 2021, 2184–2192
doi: 10.1093/jamia/ocab114
Advance Access Publication Date: 16 July 2021
Research and Applications

Research and Applications

DeepADEMiner: a deep learning pharmacovigilance pipeline for extraction and normalization of adverse drug event mentions on Twitter

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

[1]DBEI, Perelman School of Medicine, University of Pennsylvania, Philadelphia, Pennsylvania, USA, [2]Kazan Federal University, Kazan, Russia, and [3]LIACS, Leiden University, Leiden, Netherlands

Corresponding Author: Graciela Gonzalez-Hernandez, MS, PhD, DBEI, Perelman School of Medicine, University of Pennsylvania, Philadelphia, PA, USA (gragon@pennmedicine.upenn.edu)

Received 20 January 2021; Revised 20 May 2021; Editorial Decision 24 May 2021; Accepted 8 June 2021

ABSTRACT

Objective: Research on pharmacovigilance from social media data has focused on mining adverse drug events (ADEs) using annotated datasets, with publications generally focusing on 1 of 3 tasks: ADE classification, named entity recognition for identifying the span of ADE mentions, and ADE mention normalization to standardized terminologies. While the common goal of such systems is to detect ADE signals that can be used to inform public policy, it has been impeded largely by limited end-to-end solutions for large-scale analysis of social media reports for different drugs.
Materials and Methods: We present a dataset for training and evaluation of ADE pipelines where the ADE distribution is closer to the average 'natural balance' with ADEs present in about 7% of the tweets. The deep learning architecture involves an ADE extraction pipeline with individual components for all 3 tasks.
Results: The system presented achieved state-of-the-art performance on comparable datasets and scored a classification performance of $F_1 = 0.63$, span extraction performance of $F_1 = 0.44$ and an end-to-end entity resolution performance of $F_1 = 0.34$ on the presented dataset.
Discussion: The performance of the models continues to highlight multiple challenges when deploying pharmacovigilance systems that use social media data. We discuss the implications of such models in the downstream tasks of signal detection and suggest future enhancements.
Conclusion: Mining ADEs from Twitter posts using a pipeline architecture requires the different components to be trained and tuned based on input data imbalance in order to ensure optimal performance on the end-to-end resolution task.

Key words: social media mining, natural language processing, information extraction, pharmacovigilance, drug safety

INTRODUCTION

Advances in machine learning have sparked interest in the research community for developing automated methods to monitor public health using natural language processing. One particular focus area has been in discovering adverse drug events (ADEs) on social media texts, such as Twitter, or health forums, such as dailystrength.com or webmd.com among others. ADEs are negative side effects (ie, harmful and undesired reactions due to the intake of a drug/medication).[1] ADEs have been previously used interchangeably with the term adverse drug reactions (ADR). In pharmacoepidemiology,

ADR infers a causality relation between the drug and the effect. This relation is difficult to infer from nonclinical data like social media. Hence, hereafter, we prefer to use the term ADE as opposed to ADR. In this work, we present an information extraction pipeline for ADEs from Twitter by first identifying tweets that mention ADEs, then extracting the text spans of the mentions and subsequently normalizing them to the MedDRA preferred terms.

In order to conduct social media pharmacovigilance studies that require mining tweets for the presence of ADEs, the first step usually involves collecting the tweets that mention medications by using their names and variants as search terms in the Twitter API. If no other keywords are included, only a small fraction of tweets obtained would mention ADEs.[2] The reasons for this phenomenon are multifold: (1) a large proportion of drug names are mentioned in advertisements or posts by bots, (2) many drug names are ambiguous, and (3) the discourse in social media when discussing medications includes a variety of topics, hence just a few posts are mentions of drugs or, even less, ADEs.[3–6] Consequently, for effective extraction of such rare events from social media, work on this topic has often focused on the initial independent task of tweet classification so that tweets classified as containing ADEs can be analyzed by experts.[7–10]

Downstream automated extractions, such as the mentions (characterized by spans) of text of the expressed ADEs, can be performed using NER models as shown in Figure 1. Due to the higher complexity of the NER task, NERs have a lower sensitivity to identifying ADEs in tweets compared to a classifier. Additionally, NER models contain a larger number of parameters in the model compared to classifiers. Hence, NER models have typically been applied on tweets that have been previously identified by the classifier to contain an ADE. In such a configuration, the classifier acts as a filter to reduce the number of posts that do not contain an ADE. In such a pipeline architecture, the extraction performance of the NER is also linked to the performance of the classifier because ADEs in posts that were wrongly filtered out by the classifier (ie, false negatives) will never be processed by the NER. Similarly, other advanced downstream tasks, such as ADE normalization, that are performed on the extracted mentions will suffer from compounding errors. Recent advances in deep learning, especially in transfer learning, have shown that pretrained language representations like BERT and GPT-2 can achieve state-of-the-art (SOTA) performance in various information extraction tasks, such as text classification and named entity recognition, with fewer annotated examples. With such model improvements, we find it important to reevaluate if ADE classifiers continue to be an essential step in the extraction pipeline.

Related work

Methods for ADE tweet level classification have been studied extensively in various studies and shared tasks.[7,8,11,12] The ADE classification task is challenging due to the imbalanced nature of the dataset. Among tweets that mention medications, which is often a starting point for data collection, tweets that mention ADE are outnumbered 10:1 to 50:1 by tweets that do not contain ADEs.[7,8,12,13] From our preliminary analysis of datasets in shared tasks, the variability in the ratios could be largely attributed to the class of drugs being used for the study. Emerging medications are often promoted by bots as well as mentioned in news articles which overshadow firsthand reports of medication consumption by users. The precision of optimal ADE classification systems previously developed have stayed in the range of 0.45–0.65 reaching a score of 0.64 in recent shared tasks.[7,8,12] Assuming a pipeline architecture, the datasets for the NER and normalization shared tasks have thus commonly assumed an input corpus consisting of 50% of tweets positive for ADEs.

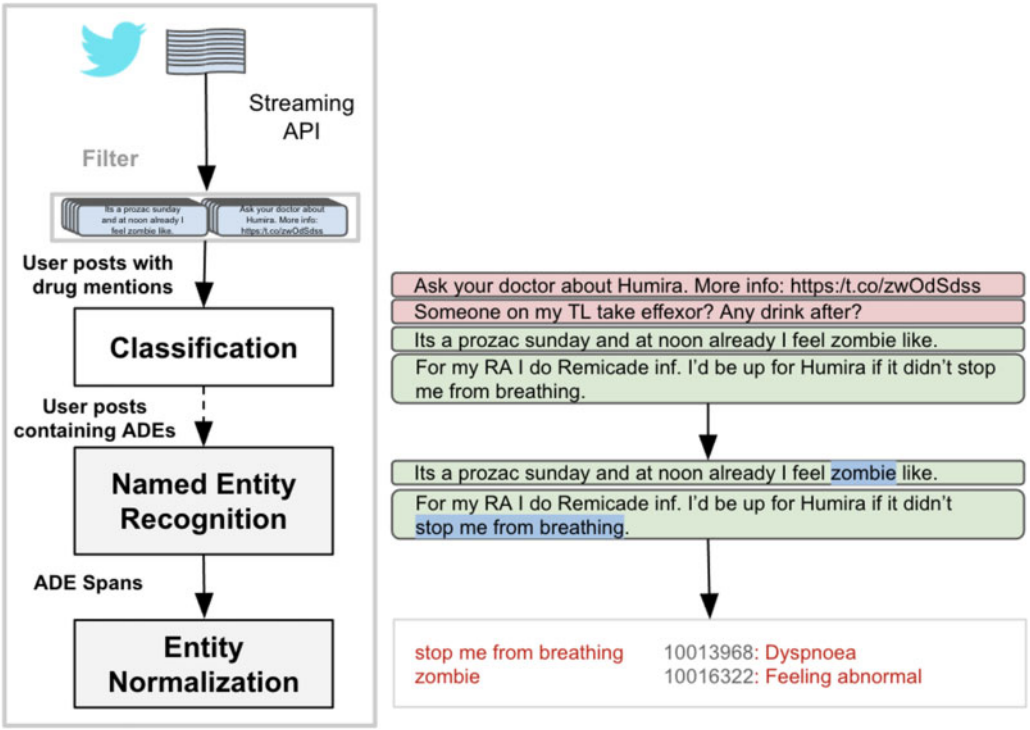

Figure 1. Typical ADE extraction pipeline from Twitter. Tweets are retrieved by either using the streaming API using drug names as keywords or searching a previously indexed database by drug name. Downstream tasks (ADE tweet classification, named entity recognition, and entity normalization) are performed serially.

However, this assumed balance for the task of ADE extraction has gone as high as 0.95 positive, in essence ignoring the ADE negative tweets in the dataset. We found that training and testing on such an extremely unbalanced dataset with mostly positive tweets creates sub-optimal models.[14,15] Here, we show that training on modified datasets under such unrealistic assumptions of ADE classification performance merely gives a false sense of the individual component's performance. Building such systems will invariably result in a large drop in performance in the end-to-end ADE resolution pipeline when executing on a dataset with the inherent imbalance of a Twitter collection.

Similarly, previous implementations of ADE normalization have often limited their target classes to the ones available only in the dataset, thereby artificially inflating the reported performance.[16–18] We find that training on only the common identifiers available in the training set or limited number of identifiers may yield better accuracy but does not allow discovery of new ADEs because target classes outside those in the training data or the datasets are not considered. Here, we demonstrate that normalization labels can be expanded using linked ontologies to yield better results and enable normalization of ADEs not available as part of the training set.

Objectives and contributions

The objective of this work is to evaluate the performance of deep learning classifiers for ADE extraction and to answer key questions on the design of ADE extraction and normalization pipelines on texts from social media, particularly Twitter. To accomplish this, we use off-the-shelf deep learning classifiers and NER tools. Following are the contributions of the work presented:

- We establish SOTA performance on an end-to-end ADE extraction and normalization pipeline.
- We make available an ADE normalizer that maps the extracted spans to MedDRA Preferred Term identifiers using the expanded vocabulary from Unified Medical Language System (UMLS).
- We make the end-to-end pipeline available to the public as an API endpoint and an online interactive tool.
- 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 evaluate the utility of including an ADE classifier as the first step of a pipeline to tackle the imbalance in the data.

The source code, binaries, and models for the systems presented here, as well as the annotated datasets, are available at https://healthlanguageprocessing.org/pubs/deepademiner/.

MATERIALS AND METHODS

Datasets

In this work, we merge datasets used in our social media for pharmacovigilance shared tasks,[8,11] resulting in a dataset of tweets mentioning 1 or more medications where only 7% of the tweets contain ADEs. The tweets were collected using the Twitter API and annotated by experts after applying preprocessing filters to remove tweets that were likely to be advertisements or from automated accounts. We refer the readers to the original articles for details regarding data collection and annotation guidelines.[2,8] The dataset contains 29 284 tweets annotated with 2765 ADE mentions. The annotated ADE mentions also contain the corresponding normalized medical term in the MedDRA ontology.[19,20] The MedDRA ontology is a standardized hierarchical medical terminology that is often used to report ADE in clinical trials. Each ADE is annotated to 1 of the 79 507 MedDRA lower-level term (LLT) identifiers. The 2765 ADE annotations contain 669 unique LLT identifiers containing 257 LLT terms in the test set that are not part of the training set. Some of the most common ADEs include *Withdrawal syndrome, Somnolence, Chronic insomnia*, and *Weight gain* with 134, 89, 59, and 58 occurrences. This dataset is split into 18 300 (62.5%) tweets with 1800 ADEs for training and 10 984 (37.5%) tweets with 965 ADEs for testing. We refer to this dataset in this work as the *HLP-ADE-v1* dataset.

For purposes of comparison with other SOTA methods on similar datasets, we also use the datasets used in Task 2 (English) and Task 3 of the social media mining for health (SMM4H 2020) shared task.[12] The training set for Task 2 contained 25 678 tweets, with 2377 (9.3%) reporting an adverse effect of a medication while the test set contains 4759 tweets, with 194 (4.1%) tweets reporting an adverse effect. The primary focus of *Task 2* was to classify tweets containing ADEs while *Task 3*'s focus was to train and evaluate span detection and normalization capabilities. The data for Task 3 contains 2806 tweets in the training set, with 1829 (65%) tweets that report an adverse effect, while the test set contains 1156 tweets with 970 (84%) that report an adverse effect. We refer to these datasets as the *SMM4H-2021* for the rest of the article.

ADE resolution pipeline components

Our approach to the ADE pipeline involves 3 components, 1 for each of the necessary tasks: (1) the ADE classifier for identifying tweets containing ADE mentions, (2) the ADE span extractor or NER for extracting ADE mentions, and (3) the ADE normalizer, which maps the extracted ADE mention to MedDRA LLT identifiers. We refer to the end-to-end pipeline as the ADE resolution task.

ADE classifier

To identify tweets that contain ADEs, each tweet in the dataset that contains at least 1 mention of an ADE is assigned the hasADE class, and the other tweets are assigned the noADE class. Based on findings from the recent SMM4H shared task, we use the transformer model RoBERTa to train a binary classifier using the Flair framework.[21,22] To deal with the class imbalance problem, the classifier is trained with varying loss weights and undersampling techniques to obtain the optimal performance with a particular focus on the *Recall* metric, which measures how sensitive the classifier is in identifying posts containing ADEs. Essentially, this classifier is used as a filter to remove tweets that are not predicted to mention ADEs; hence, we find it important to increase the sensitivity of the classifier to preserve tweets that are likely to contain ADEs for further processing by downstream components.

ADE span extractor

We used the off-the-shelf NER training framework from Flair for extracting ADE spans.[22] As part of our preliminary experiments, we examined BERT implementations across native TensorFlow, fast.ai, and Flair frameworks and found similar extraction performance among the tools.[23] As preprocessing steps, we used segtok to tokenize the tweet and label the text with the standard IOB2 (also called BIO) format for training. From the training set, 5% of the examples were held out as a development set for hyperparameter tuning. The NER model presented in this article relies on tweets represented using token

representations obtained from pretrained word embeddings. In this work, we investigate the utility and performance of multiple word embeddings for use in the NER. The token representations are used to make classification decisions for each token using sequence tagging models to indicate the presence of ADE entities. We use the bidirectional recurrent neural network-based architecture with gated recurrent units and a fully connected layer with a conditional random field on the output layer with hidden layer dimensions set to 256. We used the optimal settings to be training at 0.1 learning rate with the default optimizer based on stochastic gradient descent. The model was trained for 50 epochs, and the model with the best performance on the development set was saved for testing its performance on the test sets.

ADE normalizer

For normalizing the extracted spans to their respective MedDRA concepts, we use a classification approach where we use the text in the ADE mentions as inputs to the classifier, and their annotated LLTs are mapped to preferred terms (PTs), which are used as target classes. We thus create training examples for the classifier using the 2289 annotations available in the training set for *Supervised* training. In *Supervised* training, the target classes are limited to classes annotated in the training set, and hence the trained model will lack the capability to normalize ADEs not present in the training set.

We also create training examples using 79 507 MedDRA LLT terms and their corresponding PT identifiers as training instances. We further expand these LLT and PT terms to their synonyms using the UMLS thesaurus,[24] linking their concept unique identifiers with identifiers in other databases. This expanded the number of unique training instances to 265 255. We mapped all LLT terms to their 23 389 PTs, reducing the number of target classes and preparing them for *Unsupervised* training of the model where human annotations from the dataset are not used. Additionally, we also perform *Semi-supervised* training by using the examples from both the *Unsupervised* and *Supervised* training set for training the model as shown in Figure 2.

For normalization, we evaluate 2 classifiers. (1) The off-the-shelf FastText classifier,[25] which computes the average of token vectors using word embeddings based on presence of subwords and uses a multinomial logistic regression model with softmax layer at the output. Since the objective of normalization is to train on all available PT classes in MedDRA, we use the hierarchical softmax loss available in the FastText package for faster training. (2) We create a classifier based on BERT transformer embeddings, which incorporates context and shallow semantic information into word and documentation representations.

Experiments

Using the ADE dataset, we conduct the following experiments:

Experiment 1: ADE Classification, extraction, and normalization vs ADE resolution. We built a SOTA pipeline which employs deep learning based classifiers and NERs for detecting tweets that contain ADEs, extracting ADE mentions, and further normalizing the mentions to the MedDRA terminology. We test various NERs across word representations for ADE extraction and an entity normalization classifier for normalizing the ADE spans extracted from the NER model to the MedDRA terminology. The performance of these 3 steps is analyzed both independently and in a resolution pipeline to assess the impact of the NER and normalization on the ADE resolution performance.

Experiment 2: Effect of data imbalance. For this experiment, we first exclude the classifier from the abovementioned pipeline and create multiple datasets for the NER based on the proportion of negative tweets (noADE) in the collection in comparison to positive tweets (hasADE). We train multiple NERs and test them on the 7% positive test set to determine the impact of biased and balanced training on ADE resolution. In this experiment, we also evaluate the impact of the ADE classifier at the first step.

Evaluation

The evaluation is 2-fold. First, each component used in the pipeline is evaluated independently, followed by an end-to-end evaluation of ADE resolution. The performance of the ADE classifier is characterized using measures such as precision, recall, and F_1-score for the hasADE class. Here, precision is measured by the ratio of true positives to the sum of true positives and false positives, recall is measured by the ratio of true positives to the sum of true positives and false negatives, and finally F_1-score is measured by taking the harmonic mean of precision and recall.

The performance of the ADE extractor is measured based on the presence of overlapping spans of annotated and predicted ADE mentions in a tweet. A prediction is considered to be a true positive if any part of the predicted ADE text overlaps with the annotated ADE text. ADEs annotated that were not predicted are false negatives and ones that were predicted when there are no ADEs in the same span are considered false positives. We calculate precision, recall, and F_1-score measures for the ADE spans to compare performances between the methods.

We evaluated 4 types of embeddings: (1) traditional Glove embeddings trained on Twitter data,[26] (2) word2vec embeddings trained on Twitter data with medication mentions and health related tweets,[27] (3) FastText embeddings with enriched subword information trained on webcrawl data,[28] and (4) transformer models, namely BERT, trained on Wikipedia.[23]

The performance of the ADE normalizer is reported using the accuracy metric. Since terms in LLTs (which the corpus is annotated on) are often synonyms and spelling variants at the lowest level of granularity of the MedDRA ontology, we evaluate instead using 1 level up in the ontology, the PT level, which contains 23 389 entries (compared to 79 507 at the LLT level). Thus, if the predicted MedDRA LLT identifier maps to the same PT identifier as the annotated LLT, the prediction is considered a true positive. We perform accuracy evaluations for the normalization task across 2 subsets of the test set, (1) ADEs present in the training set and (2) ADEs not present in the training set.

We evaluate ADE resolution (the end-to-end performance of the classifier, extractor, and normalizer) on the same annotated dataset using precision, recall, and F_1-score. A prediction is considered a true positive only when the spans overlap and the normalized MedDRA PT identifiers match.

The Institutional Review Board of the University of Pennsylvania reviewed the studies for which this data was collected and deemed them exempt for human subject research under category (4) of paragraph (b) of the US Code of Federal Regulations Title 45 Section 46.101 for publicly available data sources (45 CFR §46.101(b)(4)).

RESULTS

We present results of the above tasks in the following subsections.

Experiment 1: ADE classification, extraction, and normalization vs ADE resolution

We present the performance from the normalization task in Table 1. Analyzing the results, we find that the accuracy of the BERT models, which encode contextual representation and shallow semantic information, improve substantially over the FastText model, which relies

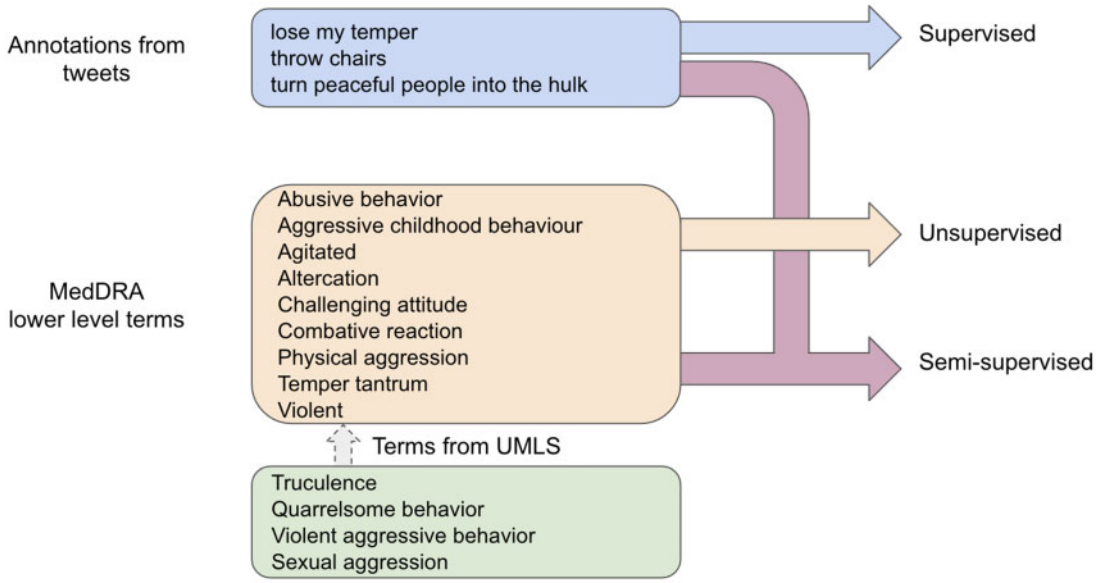

Figure 2. Normalization architecture describing the 3 methods of training based on annotations from social media and terms from MedDRA and UMLS.

Table 1. Normalization task performance on the test set operating under the assumption that extracted spans are available. Accuracy columns indicate (a) overall performance by measuring accuracy on all test spans, (b) accuracy on the span subset where MedDRA ids were part of the training set, and (c) accuracy on the span subset where MedDRA ids were only part of the test set

Method	*Configuration*	*Accuracy (overall) n = 965*	*Accuracy (training only) n = 455*	*Accuracy (test only) n = 510*
FastText	Unsupervised	0.414	0.425	0.402
	Supervised	0.495	0.442	0
	Semisupervised	0.521	0.551	0.411
BERT	Unsupervised	0.441	0.447	0.415
	Supervised	0.590	0.653	0
	Semisupervised	0.612	0.638	0.497

on bag-of-words and n-grams to perform the normalization task. For the given dataset containing about 1300 annotations of MedDRA identifiers, supervised learning outperformed unsupervised learning by a margin of 8 to 14 percentage points. However, it is surprising to find that even when the number of classes exceed 23 000, unsupervised training based on MedDRA text entries provides an accuracy of 0.414 with FastText and 0.441 with BERT word representations.

Overall, the classification methods that included both supervised labels and unsupervised labels performed better than unsupervised methods and systems trained only on supervised labels. As the test set contains ADE preferred terms that are not present in the training set, we find that the semisupervised approach performs better and allows for discovering ADEs not present in the training set.

Table 2 shows the performance of various language representation techniques for the ADE extraction task when trained on the full dataset in the absence of a classifier. We find that the NER that uses the BERT representations equipped with an additional layer of bidirectional gated recurrent units and a conditional random field layer obtains the best performance when trained on the full training dataset. However, we suspected that the large class imbalance may have had an impact on the NER performance, hence we proceeded to run multiple folds of training data with undersampling techniques to determine the optimal ratio of negative to positive tweets containing ADEs.

Experiment 2: Effect of data imbalance

Firstly, we compare the performance of the classifier and the NER for identifying tweets that contain ADEs. Since the loss function of the classification tasks and NER tasks are defined differently, we naturally expect the NER to perform lower in the classification task. The models that were trained with 5 times as many negative tweets compared to the positive tweets were found to have the optimal performance as shown in Figure 3. We also find that lowering the threshold of classification to 0.15 from 0.5 for the undersampled classifier model further increases the performance of the classifier as shown in Figure 4. Comparing classification performances, we found that the ADE classifier fine-tuned using the RoBERTa model (F_1-score = 0.63) outperforms the NER (F_1-score = 0.41) in identifying tweets containing ADEs by about 22 percentage points. The best performance of the classifier was obtained at a probability threshold where both precision and recall were around 0.63. We find that greater F_1-score by the classifier allows for improvement in the NER's performance and overall ADE resolution pipeline.

The performance of the NER across varying ratios of hasADE/noADE using FastText embeddings is shown in Figure 5. Observing the figure, we can see that when training the NER on its own, the peak performance of the FastText model occurs for ratios in the range of 1–2. We made similar observations for the BERT model. Based on these findings, we retrained the classifier and NER and evaluated the model across the presented dataset and similar data-

Table 2. ADE span extraction performance using overlapping precision, recall, and F1-scores when trained on the full dataset in the absence of a classifier

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

sets published as part of the SMM4H workshop.[12] We present the results in Table 3. The systems presented in this work improves over previous SOTA systems.

Overall, among the 510 annotations that contained LLT ids that were present in the test set but not the training set, DeepADEMiner successfully classified, extracted, and normalized 116 ADE annotations at the PT id level. They included ADEs such as *heartburn* (10013946: Dyspepsia), *wooziness* (10013573: Dizziness), *spaced out* (10016322: Feeling abnormal), *hiccups* (10020039: Hiccups) and *mouth numb* (10057371: Hypoaesthesia oral). Despite the training set not containing any of the above examples of ADE mentions or their LLT ids, the NER had the capability to extract the mention and semisupervised training on MedDRA ontology terms, with terms integrated from UMLS, and had the capability to classify them correctly to their correct PT ids.

To measure the generalizability of DeepADEMiner for extraction and normalization of ADEs, we split the annotations in the test dataset into 2 categories based on the presence or absence of annotated MedDRA LLT ids or ADE spans in the training dataset. Among the 455 ADEs in the test dataset that had their corresponding MedDRA LLT ids in the training dataset, 165 were extracted and normalized correctly (Recall = 0.363). Among the 510 ADEs in the test dataset that did not have their corresponding MedDRA LLT ids in the training dataset, 116 were extracted and normalized correctly (Recall = 0.227). Among the 340 ADEs in the test dataset that had their corresponding text spans in the training dataset, 180 were extracted and normalized correctly (Recall = 0.529). Among the 625 ADEs in the test dataset that did not have their spans matching any ADEs in the training dataset, 101 were extracted and normalized correctly (Recall = 0.162). Results from the evaluation of DeepADEMiner's normalization model suggests that using semisupervised methods is beneficial in identifying ADEs not present in the training set.

In this work we performed all experiments and evaluated the models on a test dataset where only 7% of tweets contained ADEs. DeepADEMiner achieves a resolution performance of Precision = 0.41, Recall = 0.29, and F_1-score = 0.34. However, we recognize that the datasets may contain a higher level of imbalance where the proportion of tweets containing ADEs are as low as 2%.[7,8,12,13] To estimate the performance of DeepADEMiner on such datasets, we created variants of the *HLP-ADE-v1* test set where the presence of tweets with ADE was 5% and 2% by randomly replacing tweets containing ADEs with tweets not containing any ADEs from the *SMM4H-2020* dataset. We evaluated the performance of DeepADEMiner on these datasets and found that, on the 5% dataset, DeepADEMiner achieves a Precision = 0.303, Recall = 0.285, and F_1-score = 0.305. On the 2% dataset, the performance of DeepADEMiner further reduces to Precision = 0.212, Recall = 0.251, and F_1-score = 0.230. This shows that when the proportion of tweets containing ADEs to tweets without ADEs is further reduced from 7% to 2%, we can expect the performance of DeepADEMiner to drop by approximately 11 percentage points, which may impact the quality of extracted ADEs for manual analysis and systematic studies. Despite the performance of the end-to-end pipeline, we believe that the presented tool holds tremendous utility for social media pharmacovigilance.

DISCUSSION

In this article, we evaluate extraction and normalization of ADEs on realistic, imbalanced data. Our deep learning architecture achieves a span extraction performance of F_1= 0.44 and an end-to-end performance (ie, classification, extraction, and normalization) of 0.34. Inclusion of a higher proportion of tweets that do not contain ADEs during training improves the F_1score of ADE span extraction when

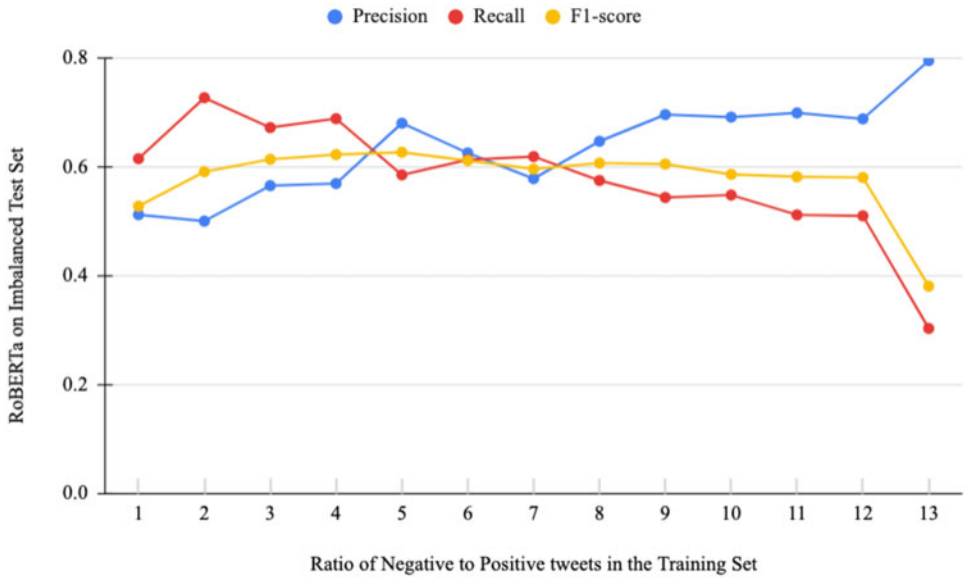

Figure 3. The chart shows how the variation in proportion of tweets in noADE and hasADE classes affects the performance of the ADE classifier.

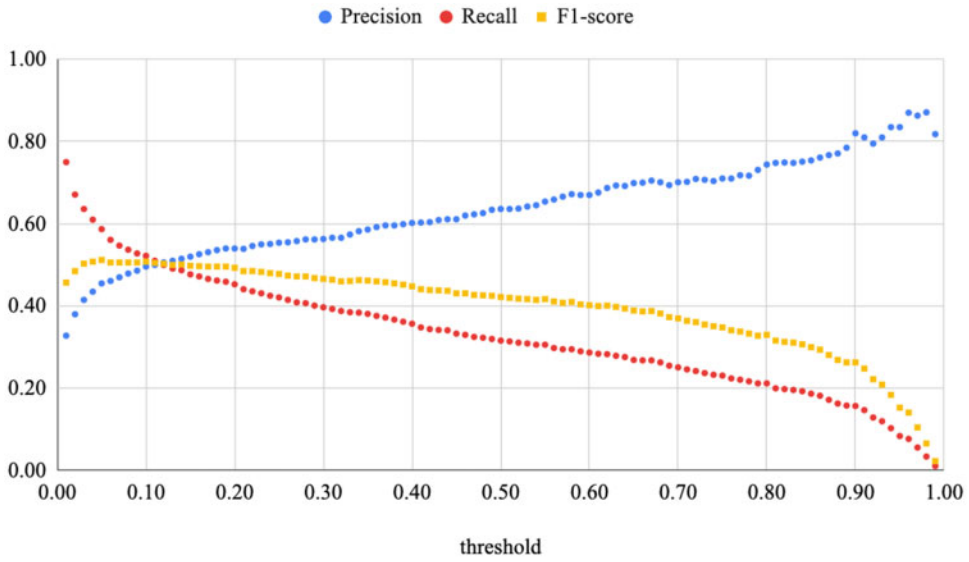

Figure 4. The chart shows how the varying threshold of the classifier affects the classification performance on the development set. For this experiment we used the undersampled classifier where the ratio of noADE to hasADE is 5.

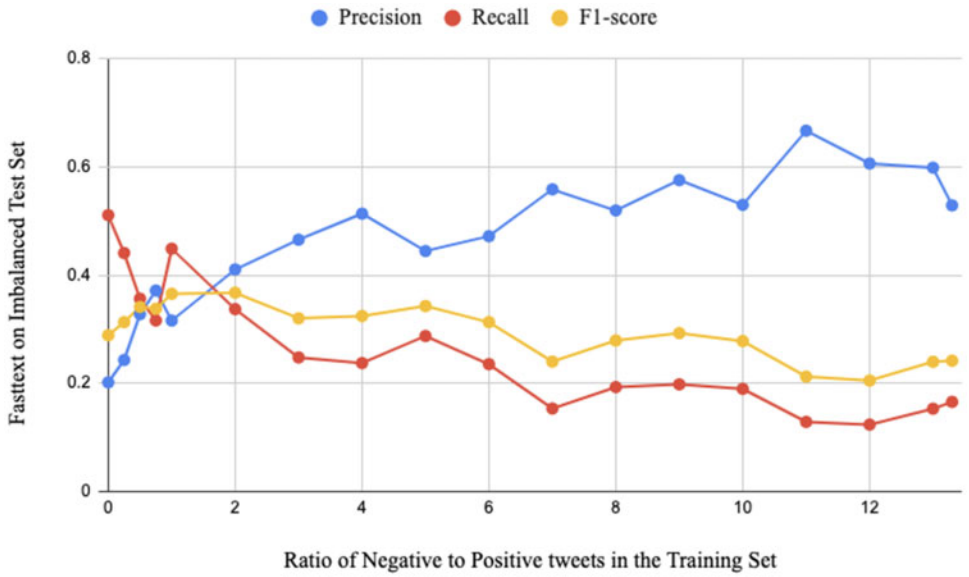

Figure 5. The chart shows how the variation in proportion of tweets in noADE and hasADE classes affects the performance of the ADE span extraction system suggesting that inclusion of tweets that do not contain ADEs improves the overall F1-measure of the NER when this ratio is in the range of 1–5 and decreases substantially with further inclusion of noADE tweets.

the ratio of negative to positive tweets in the training set is in between 1 and 2, but performance decreases after the negative tweets outnumber positive tweets by 4–5 times. In previous work, classifiers that categorized tweets that contained ADEs and those that did not were employed to tackle the data imbalance. Despite advances in the NER models, we find that adding an ADE classifier as a first step in the pipeline is beneficial.

These findings inform the optimal setup for an end-to-end ADE resolution pipeline. The first step is an ADE classifier that is trained by undersampling the ADE negative class such that the ratio of ADE negative to ADE positive tweets is in between 1:5, reduced from the original ratio of 1:13. Despite the undersampling methods used in training, we find that we can further improve the sensitivity of the classifier by lowering the probability threshold for the hasADE class from 0.5 to 0.15 where a probability of 1.0 indicates hasADE and 0.0 indicates noADE. The classifier is followed by an ADE extraction model that outputs the span of ADE mentions from tweets that are labeled as ADE positive by the classifier. The ADE extractor is trained using undersampling techniques similar to the classifier with a ratio of negative to positive tweets between 1 and 2. The ADE mention spans are used by the normalizer for classifying the ADEs extracted into the appropriate MedDRA PT id. The normalizer is trained using MedDRA's lower-level terms expanded using UMLS CUIs. We observe that all 3 components achieve the best performance when used with BERT encoded sentences and phrases. We maintain that the undersampling optimization strategies chosen

Table 3. Performance comparison of the components introduced in this work with state-of-the-art (SOTA) implementations and datasets. It is important to note that SMM4H datasets for NER and resolution used a balanced corpus, while the HLP-ADE-v1 corpus introduced in this work is an imbalanced corpus. The DeepADEMiner tool improves over existing SOTA scores on both corpora

Task	Corpus	State-of-the-Art P/R/F_1	DeepADEMiner P/R/F_1
Classification	SMM4H-2020	0.62/0.65/0.64	0.63/0.67/0.65
	HLP-ADE-v1	–	0.61/0.64/0.63
NER	SMM4H-2020	0.79/0.72/0.75	0.82/0.76/0.78
	HLP-ADE-v1	–	0.53/0.38/0.44
Resolution	SMM4H-2020	0.48/0.45/0.46	0.52/0.49/0.51
	HLP-ADE-v1	–	0.41/0.29/0.34

should be restricted to only the training set, and all models should be evaluated on the unmodified test set where the ratio of tweets containing ADEs to tweets not containing ADEs remains unchanged.

For ADE span extraction, we tested all the word representations and found the performance of the Glove twitter embeddings to be 4 percentage points lower than average, compared to FastText and BERT embeddings. We found that FastText embeddings performed at par with BERT embeddings despite having fewer parameters in the model. For the experiments proposed previously, we report scores from the BERT embeddings, as the performance of the NER was found to be the best under that configuration. For deploying ADE span extraction into any social media pharmacovigilance system, it is important to consider the data imbalance in the posts retrieved. From our findings, since the optimal training ratio of noADE to hasADE for the NER is in between 1 and 2, using a classifier with a precision in between 0.33 and 0.50 for the hasADE class would be considered ideal.

Future work

We present in this article a pipeline approach to solving the difficult task of ADE resolution on Twitter. One of the limitations in this approach is that we assume that adverse effects that are collocated in the tweet with drug mentions are adverse drug events. However, this may not be true in cases where more than 1 drug or adverse effect is mentioned in the tweet, and many pairs may not have the adverse relation between them. To tackle this, we intend to expand the dataset to include relation extraction annotations between drug and adverse effects.

Although the performance of the ADE resolution pipeline appears low compared to similar pipeline approaches in other domains, such as clinical data or drug labels, we find that this is largely due to fewer overlaps in ADE mentions in the training and test set in Twitter data.[29] While Twitter data are considerably more difficult to mine for ADE due to inherent noise and vagueness in the tweets, we find that additional annotations and multicorpus training may help improve the NER and normalization system's performance to further improve the end-to-end resolution pipeline performance. From our analysis of the annotated data, we may observe variation in the performance of ADE resolution pipelines because the proportion of tweets that are positive and negative to ADE mentions vary by medications and over time, as more users and organizations adopt the social media platforms for networking and outreach.

Using the presented ADE resolution pipeline, DeepADEMiner, we intend to pursue specific case studies to fully assess the value of Twitter data, in particular, and social media data, in general, for pharmacovigilance.

CONCLUSION

Approaches to mining ADEs from Twitter and social media in general that do not take into account the 'natural balance' of the datasets that serve as input can easily over-estimate the expected performance. We find that managing and dealing with data imbalance is key to obtaining optimal performance across the components in a pipeline architecture. Mining ADEs from Twitter posts using a pipeline architecture requires the different components to be trained and tuned based on input data imbalance in order to ensure optimal performance on the end-to-end ADE resolution task.

ACKNOWLEDGMENTS

MedDRA, the Medical Dictionary for Regulatory Activities, terminology is the international medical terminology developed under the auspices of the International Council for Harmonization of Technical Requirements for Pharmaceuticals for Human Use (ICH). MedDRA trademark is registered by IFPMA on behalf of ICH. We are very grateful to the annotation team led by Karen O'Connor, who annotated most of the tweets used in this study.

FUNDING

The work at University of Pennsylvania was supported by the National Institutes of Health (NIH) National Library of Medicine (NLM) grant R01LM011176 awarded to GG. The content is solely the responsibility of the authors and does not necessarily represent the views of the NIH or the NLM. The work at Leiden University was supported by SIDN fonds awarded to AD. The work at Kazan Federal University on BERT-based models and manuscript was supported by the Russian Science Foundation [grant number 18-11-00284].

AUTHOR CONTRIBUTIONS

AM, ZM, and AD performed the experiments, and AM created the overall pipeline, aggregated the results, and created the initial draft of the article. SV, ET, AD, DW, IA, and GGH added substantial sections of the article and performed repeated reviews.

DATA AVAILABILITY STATEMENT

The pipeline binaries, models, source code, and datasets used in this article will be made available on the website: https://healthlanguageprocessing.org/pubs/deepademiner/.

CONFLICT OF INTEREST STATEMENT

None declared.

REFERENCES

1. Edwards IR, Ralph Edwards I, Aronson JK. Adverse drug reactions: definitions, diagnosis, and management. *Lancet* 2000; 356 (9237): 1255–9.
2. Nikfarjam A, Sarker A, O'Connor K, *et al.* 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. Sarker A, Ginn R, Nikfarjam A, *et al.* Utilizing social media data for pharmacovigilance: a review. *J Biomed Inform* 2015; 54: 202–12.

4. Sloane R, Osanlou O, Lewis D, *et al*. Social media and pharmacovigilance: a review of the opportunities and challenges. *Br J Clin Pharmacol* 2015; 80 (4): 910–20.
5. Tricco AC, Zarin W, Lillie E, *et al*. Utility of social media and crowd-intelligence data for pharmacovigilance: a scoping review. *BMC Med Inform Decis Mak* 2018; 18 (1): 38.
6. Pappa D, Stergioulas LK. Harnessing social media data for pharmacovigilance: a review of current state of the art, challenges and future directions. *Int J Data Sci Anal* 2019; 8 (2): 113–35.
7. Weissenbacher D, Sarker A, Paul MJ, *et al*. Overview of the third Social Media Mining for Health (SMM4H) shared tasks at EMNLP 2018. In: *Proceedings of the 2018 EMNLP Workshop SMM4H: 3rd Social Media Mining Health. Applications Workshop & Shared Task*; October 31, 2018; Brussels, Belgium.
8. Weissenbacher D, Sarker A, Magge A, *et al*. Overview of the fourth Social Media Mining for Health (SMM4H) shared tasks at ACL 2019. In: *Proceedings of the Fourth Social Media Mining Health. Applications Workshop & Shared Task*; August 2, 2019; Florence, Italy.
9. Sarker A, Gonzalez G. Portable automatic text classification for adverse drug reaction detection via multi-corpus training. *J Biomed Inform* 2015; 53: 196–207.
10. Wang C-K, Dai H-J, Wang F-D, *et al*. Adverse drug reaction post classification with imbalanced classification techniques. In: *2018 Conference on Technologies and Applications of Artificial Intelligence (TAAI)*; Nov 30 2018; Taichung, Taiwan.
11. Sarker A, Belousov M, Friedrichs J, *et al*. Data and systems for medication-related text classification and concept normalization from Twitter: insights from the Social Media Mining for Health (SMM4H)-2017 shared task. *J Am Med Inform Assoc* 2018; 25 (10): 1274–83.
12. Gonzalez-Hernandez G, Klein AZ, Flores I, *et al*., eds. *Proceedings of the Fifth Social Media Mining for Health Applications Workshop & Shared Task*. Barcelona, Spain: Association for Computational Linguistics; 2020. https://www.aclweb.org/anthology/2020.smm4h-1.0.
13. Dietrich J, Gattepaille LM, Grum BA, *et al*. Adverse events in twitter-development of a benchmark reference dataset: results from IMI WEB-RADR. *Drug Saf* 2020; 43 (5): 467–78.
14. Magge A, Sarker A, Nikfarjam A, *et al*. Comment on: "Deep learning for pharmacovigilance: recurrent neural network architectures for labeling adverse drug reactions in Twitter posts. *J Am Med Inform Assoc* 2019; 26 (6): 577–9. doi:10.1093/jamia/ocz013
15. Weissenbacher D, Sarker A, Klein A, *et al*. Deep neural networks ensemble for detecting medication mentions in tweets. *J Am Med Inform Assoc* 2019; 26 (12): 1618–26.
16. Miftahutdinov Z, Alimova I, Tutubalina E. KFU NLP team at SMM4H 2019 tasks: Want to extract adverse drugs reactions from tweets? BERT to the rescue. In: *Proceedings of the Fourth Social Media Mining for Health Applications (# SMM4H) Workshop & Shared Task*; August 2, 2019; Florence, Italy.
17. Limsopatham N, Collier N. Normalising medical concepts in social media texts by learning semantic representation. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 Long Papers); August 7, 2016; Berlin, Germany.
18. Miftahutdinov Z, Tutubalina E. Deep neural models for medical concept normalization in user-generated texts. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*; August 2, 2019; Florence, Italy.
19. Mozzicato P. MedDRA: an overview of the dictionary for regulatory activities. *Pharmaceut Med* 2009; 23 (2): 65. doi:10.1007/bf03256752.
20. Brown EG, Wood L, Wood S. The Medical Dictionary for Regulatory Activities (MedDRA). *Drug Saf* 1999; doi:10.2165/00002018-199920020-00002.
21. Liu Y, Ott M, Goyal N, *et al*. Roberta: a robustly optimized BERT pretraining approach. *arXiv Prepr arXiv190711692*; 2019.
22. Akbik A, Blythe D, Vollgraf R. Contextual string embeddings for sequence labeling. In: *Proceedings of the 27th International Conference on Computational Linguistics*; August 20, 2018; Santa Fe, New Mexico, USA.
23. Devlin J, Chang M-W, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers); June 2019: 4171–86; Minneapolis, MN.
24. Bodenreider O. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Res* 2004; 32 (90001): 267D–270.
25. Joulin A, Grave E, Bojanowski P, *et al*. Bag of tricks for efficient text classification. In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Association for Computational Linguistics; April 3, 2017: 427–31; Valencia, Spain.
26. Pennington J, Socher R, Manning CD. Glove: global vectors for word representation. In: *EMNLP* 2014: 1532–43.
27. Tutubalina EV, Miftahutdinov ZS, Nugmanov RI, *et al*. Using semantic analysis of texts for the identification of drugs with similar therapeutic effects. *Russ Chem Bull* 2017; 66 (11): 2180–9.
28. Bojanowski P, Grave E, Joulin A, *et al*. Enriching word vectors with subword information. *TACL* 2017; 5: 135–46.
29. Elena T, Kadurin A, Miftahutdinov Z. Fair Evaluation in Concept Normalization: a Large-scale Comparative Analysis for BERT-based Models. In: *Proceedings of the 28th International Conference on Computational Linguistics*; December 8, 2020; Barcelona, Spain (Online).