Detection of Adverse Drug Reaction from Twitter Data

Fuchiang Tsui, PhD[1,2,3], Lingyun Shi, MS[1], Victor Ruiz, MS[1], Amie D. Barda, MS[1], Ye Ye, MS[1,2], Diyang Xue, MS[1,2], Fan Mi, MS[1], and Utkars Jain, MS[3]
[1]Department of Biomedical Informatics, University of Pittsburgh, Pittsburgh, PA, USA
[2]Intelligent Systems Program, University of Pittsburgh, Pittsburgh, PA, USA
[3]Department of Bioengineering, University of Pittsburgh, Pittsburgh, PA, USA

Abstract

In response to the challenges set forth by the 2nd Social Media Mining for Health Applications Shared Task 2017, we describe a framework to automatically detect Twitter tweets with mentioned adverse drug reactions (ADRs). We used a dataset with 10822 annotated tweets provided by the event organizers to develop a framework comprised of a pre-processing module, 6 feature extraction modules, and one predictive modeling module with 5 models (K2 Bayesian network, naïve Bayes, decision tree, random forest, and support vector machines). To evaluate our framework, we employed a blind test dataset with 9961 tweets provided by the event organizer. The area under the ROC curve (AUC) from the K2 Bayesian network model was 0.74 (95% C.I. 0,721-0.759) and F-Measure ranged from 0.339 to 0.342. The described framework in this paper demonstrated a potential public health surveillance tool for ADR surveillance from Twitter tweets.

I. Introduction

The 2nd Social Media Mining for Health Applications Shared Task (SMM4H) 2017 put forth three competition challenge tasks: Task 1: Automatic classification of adverse drug reaction mentioning posts—binary classification, Task 2: Automatic classification of posts describing medication intake—three-class classification, and Task 3: Automatic normalization of adverse drug reaction mentions. In this paper, we describe a framework to address the Task 1 challenge for classification of adverse drug reaction mentioning Twitter tweets.

Adverse drug reactions (ADRs) impose clinical and economic burdens in the U.S. The Federal Drug and Administration (FDA) shows the patients with ADRs have double length of stay and mortality than those of non-ADR patients.[1] Moreover, up to 39% of ADRs in pediatric inpatients can be life-threatening or fatal.[2] A study estimated that between 32% and 69% of drug-related admissions could be potentially preventable.[3] In the U.S., the cost of ADRs may be up to $136 billion annually.[4]

With 328 million monthly active users generating 500 million tweets per day in 2017 and 80% of users posting tweets about themselves,[5] Twitter provides a potential channel for ADR surveillance.

We hypothesize that our proposed pipeline process and predictive models can identify individual tweets with ADR.

II. Methods

The training dataset provided by SMM4H 2017 originally comprised of total 15,667 tweet identifiers with annotated ADR results from two batches: each with 10,822 and 4845 identifiers respectively; after employing the python script provided by the event organizer, we only retrieved 10,281 tweets for the training dataset. The blind test dataset released 7 days prior to the competition deadline comprised of 9,961 tweets. All the datasets were annotated and provided by the event organizer.

II.1 Data Preprocessing

Figure 1 summarizes our proposed framework, which contained a data preprocessing module, and 6 feature extraction modules, and one predictive modeling module. We first preprocessed each tweet in the training dataset in 7 steps: (1) lowercase text conversion, (2) sentence partition, (3) stop words removal, (4) tokenization, (5) spelling correction, (6) lemmatization and (7) part-of-speech (POS) tagging. In the stop words removal step, we removed commonly used words that are typically ignored by a search engine, such as “the”, “a”, “at”, etc. In the (Twitter-specific) tokenization step, we parsed Twitter-specific tokens such as retweet flag (RT), @USERNAME, Hashtag (#), URL and emoticon, and used them as features for each tweet. We used the Natural Language Processing Toolkit[6] (NLTK) to perform the data preprocessing.

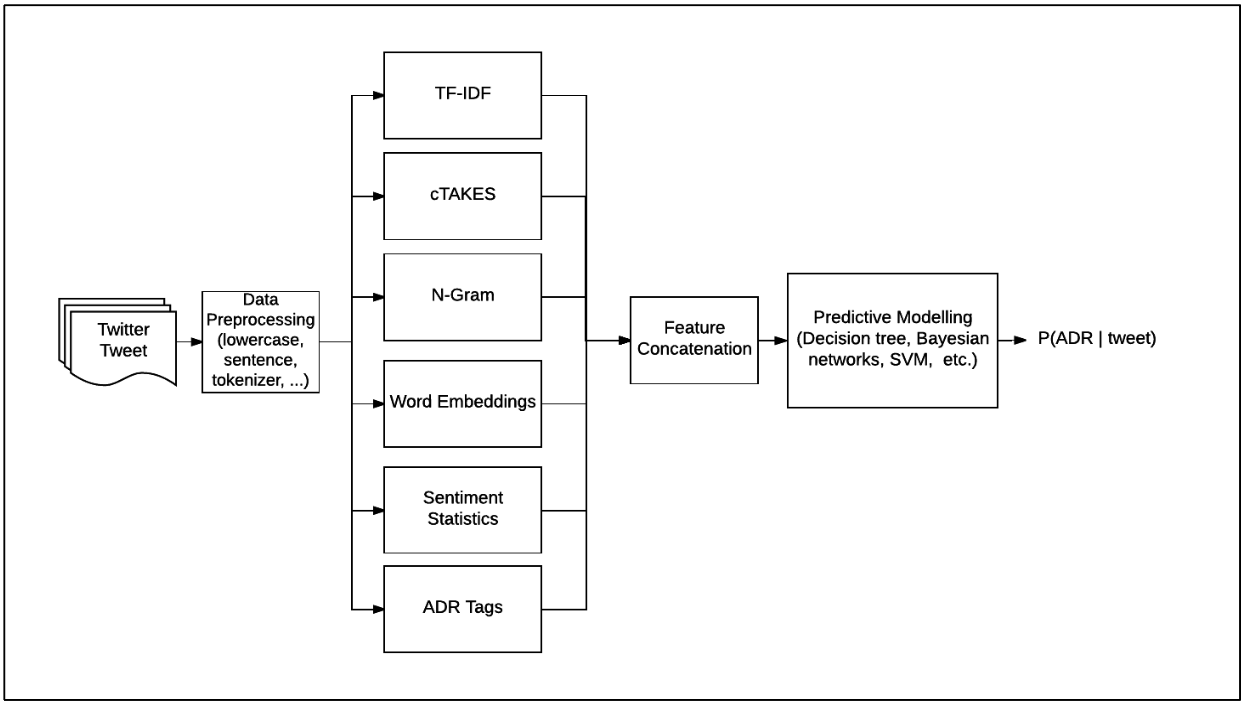
Twitter Tweet
Data Preprocessing (lowercase, sentence, tokenizer, ...)
TF-IDF
cTAKES
N-Gram
Word Embeddings
Sentiment Statistics
ADR Tags
Feature Concatenation
Predictive Modelling (Decision tree, Bayesian networks, SVM, etc.)
P(ADR | tweet)

Figure 1. Framework for classification of adverse drug reaction mentioning Twitter tweets.

II.2 Feature Modules

We employed six feature extraction modules: N-grams, term frequency-inverse document frequency (TF-IDF), word embeddings, and sentiment statistics.

II.2.1 N-grams

We used unigram (single word), bi-gram (two words) and tri-gram (three words) as features for predictive modeling. For example, single words like “sick” and “weight” belong to unigram and “Cipro pills” containing two words is a bigram.

II.2.2 Term Frequency-Inverse Document Frequency

TF-IDF score measures how important a term, e.g., word, is to a document (tweet) in a corpus (training dataset). The following equation defines TF-IDF.

$$TFIDF(t,d,D) = TF(t,d) * \log\frac{|D| + 1}{DF(t,D) + 1},$$

where *TF(t,d)* denotes the term frequency (number of times that a term *t* appears in a tweet *d*), |*D*| is the total number of tweets, and *DF(t, D)* is the document frequency (number of tweets that contains the term *t* in all tweets D). Instead of using words in the above equation, we used hashed terms for computational efficiency. We used the vector of the hashed terms from each tweet’s as a feature vector.

II.2.3 Word Embeddings

We employed the Apache Spark Word2Vec[7] to obtain a word embeddings matrix from the training dataset. We used Skip-gram model with softmax activation function in a neuron. The Skip-gram model learns word vector representations from sentences by maximizing the average log-likelihood of a word w_t in a sentence based on conditional independence assumption as shown below:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{j=-k}^{j=k}\log p\left(w_{t+j}\middle|w_t\right),$$

where k is the size of a pre-defined training window, T is the total number of words within a sentence, and the conditional probability $\mathrm{p}(w_i|w_j)$ is determined based on the softmax model.

For each tweet, we computed an averaged word vector based on the word embeddings matrix, which becomes a feature vector.

II.2.4 Sentiment statistics

We employed Stanford NLP toolkit[8,9] to annotate each sentence's sentiment level as one of 5 levels in a tweet: very negative, negative, neutral, positive or very positive. From the annotation, we generated 6 sentiment features for each tweet: 1) the count of negative sentences annotated with negative and very negative sentiment levels, 2) the ratio between the numbers of negative sentences and all sentences, 3) the count of sentences annotated with neutral sentiment level, 4) the ratio between the numbers of neutral sentences and all sentences, 5) the count of positive sentences annotated with positive and very positive sentiment levels, 6) the ratio between the numbers of positive sentences and all sentences.

II.2.5 cTAKES findings

We employed cTAKES[10], a medical natural language processing application, to extract symptoms and findings from clinical narratives. We further customized cTAKES with 2017 US Edition of SNOMED CT[11] as a clinical-term look-up dictionary. We identified 10 annotation types such as drug, disorder, finding, procedure, lab, etc.

II.2.6 ADR tags

We employed ADRMine[12], a social-media-specific named entity recognition (NER) application, to identify ADR mentions in a tweet. Any non-negative ADR mentions in the training dataset were included in our feature set.

II.3 Predictive Modeling

We used five machine learning algorithms to build predictive models and validated the models through nested 10-fold cross validation. All the features were obtained from the feature modules stated in Section II.2. For each fold, we applied information gain first followed by correlation-based feature selection. We then used K2, Naïve Bayes (NB), decision tree (DT), random forest (RF), and SVM algorithms to build predictive models.

We identified three different methods to determine probability thresholds. The first method is *cross-validation threshold*, which used the average threshold that optimized F1-score during cross-validation testing. The second method is *prevalence threshold*, which calculated the threshold that would produce the same class prevalence observed in the training set. The third method is *training threshold*, which calculated the threshold that optimized F1-score in the entire training dataset.

The evaluation metrics for predictive models are F-measure (also known as F1-measure), precision, recall, and the area under the ROC curve (AUC).

III. Results

Compared to other models, the K2 model had the highest average AUC (0.844) and F1-score (0.469) across 10 test folds within the 10-fold cross validation (CV). Table 1 lists predictive results from the 5 algorithms within the 10-fold CV. DT had the best recall (0.641) and SVM (0.62) had the best precision.

Table 1. Prediction performance within 10-fold cross validation from 5 algorithms: K2, Naïve Bayes (NB), Decision Tree (DT), Random Forest (RF), and Support Vector Machines (SVM).

Algorithm	AUC	F-Measure	Precision	Recall
K2	**0.844**	**0.469**	0.431	0.519
NB	0.828	0.446	0.399	0.519
DT	0.783	0.384	0.310	**0.641**
RF	0.756	0.388	0.326	0.504

SVM	0.581	0.269	0**.620**	0.174

The final K2 model comprised 26 nodes from unigram (9 nodes), bi-gram (2 nodes), tri-gram (1 node), cTAKES (8 nodes), word embeddings (1 node), and ADR tags (5 nodes).

Based on Table 1, we used K2 model for the final test (blind) dataset provided by the organizer released a week before the predictive result submission. Table 2 summarizes the predictive performance in the test dataset. The AUC of the K2 model was 0.74 (95% C.I. 0,721-0.759). The K2 model with the *training threshold* method had the best F-measure (0.342) and the *prevalence threshold* method had the best recall (0.394).

Table 2. K2 Prediction Performance in the Test (blind) Dataset.

Algorithm	AUC	F-Measure	Precision	Recall
K2-CV	0.74	0.341	0.333	0.35
K2-Prevalence	0.74	0.339	0.298	**0.394**
K2-Training	0.74	**0.342**	**0.336**	0.348

IV. Conclusion

In this study, we built and evaluated a framework to identify individual tweets with mentioned ADR. The framework comprises of preprocessing module, feature extraction modules, and predictive modeling module. The results demonstrated a potential public health surveillance tool for ADR surveillance from Twitter tweets.

References

1. U.S. Food and Drug Administration. Costs associated with ADRs. https://www.fda.gov/drugs/developmentapprovalprocess/developmentresources/druginteractionslabeling/ucm114949.htm. Accessed October 5, 2017.
2. Impicciatore P, Choonara I, Clarkson A, Provasi D, Pandolfini C, Bonati M. Incidence of adverse drug reactions in paediatric in/out-patients: a systematic review and meta-analysis of prospective studies. *Br J Clin Pharmacol*. 2001;52(1):77-83. doi:10.1046/j.0306-5251.2001.01407.x.
3. Sultana J, Cutroneo P, Trifirò G. Clinical and economic burden of adverse drug reactions. *J Pharmacol Pharmacother*. 2013;4(Suppl 1):S73-7. doi:10.4103/0976-500X.120957.
4. Bootman J, Johnson JA. Drug-related morbidity and mortality: A cost-of-illness model. *Arch Intern Med*. 1995;155(18):1949-1956. doi:10.1001/archinte.1995.00430180043006.
5. Naaman M, Boase J, Lai C-H. Is it really about me? In: *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work - CSCW '10*. ; 2010:189. doi:10.1145/1718918.1718953.
6. Bird S, Klein E, Loper E. *Natural Language Processing with Python*. Vol 43.; 2009. doi:10.1097/00004770-200204000-00018.
7. Meng X, Bradley J, Yavuz B, et al. [seminal] MLlib: Machine Learning in Apache Spark. *J Mach Learn Res*. 2016;17:1-7. http://arxiv.org/abs/1505.06807.
8. Manning C, Surdeanu M, Bauer J, Finkel J, Bethard S, McClosky D. The Stanford CoreNLP Natural Language Processing Toolkit. In: *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. ; 2014:55-60. doi:10.3115/v1/P14-5010.
9. Socher R, Perelygin A, Wu J. Recursive deep models for semantic compositionality over a sentiment treebank. *Proc …*. 2013:1631-1642. doi:10.1371/journal.pone.0073791.
10. Savova GK, Masanz JJ, Ogren P V, et al. Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): architecture, component evaluation and applications. *J Am Med Inf Assoc*. 2010;17(5):507-513. doi:10.1136/jamia.2009.001560.
11. NIH-NLM. SNOMED Clinical Terms® (SNOMED CT®). *NIH-US Natl Libr Med*. 2015.
12. 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 Informatics Assoc*. 2015;22(3):671-681. doi:10.1093/jamia/ocu041.