

Original Research Paper

Extended Trigger Terms for Extracting Adverse Drug Reactions in Social Media Texts

¹Rami Naim Mohammad Yousef, ²Sabrina Tiun and ³Nazlia Omar

¹Palestine Technical University (PTUK), Tulkarm, Palestine

^{2,3}Center for Artificial Intelligence Technology (CAIT), Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM), 43600, Bangi, Selangor, Malaysia

Article history

Received: 6-05-2019

Revised: 11-06-2019

Accepted: 29-06-2019

Corresponding Author:

Rami Naim Mohammad Yousef
Palestine Technical University
(PTUK), Tulkarm, Palestine
Email: p88515@siswa.ukm.edu.my

Abstract: Adverse Drug Reaction (ADR) is a disorder caused by taking medications. Studies have addressed extracting ADRs from social networks where users express their opinion regarding a specific medication. Extracting entities mainly depends on specific terms called trigger terms that may occur before or after ADRs. However, these terms should be extended, especially when examining multiple representation of N-gram. This study aims to propose an extension of trigger terms based on the multiple representation of N-gram. Two benchmark datasets are used in the experiments and three classifiers, namely, support vector machine, Naïve Bayes and linear regression, are trained on the proposed extension. Furthermore, two document representations have been utilized including Term Frequency Inverse Document Frequency (TFIDF) and Count Vector (CV). Results show that the proposed extended trigger terms outperform the baseline by achieving 88% and 69% of F1-scores for the first and second datasets, respectively. This finding implies the effectiveness of the proposed extended trigger terms in terms of detecting new ADRs.

Keywords: Adverse Drug Reaction, Trigger Terms, Feature Extraction, Support Vector Machine, Linear Regression

Introduction

The exponential development of social networks has affected several domains of interests, such as marketing, business and arts. For instance, the medical domain has gained interest among social media users who can give their opinions about this domain (Denecke and Deng, 2015). An example of such opinions is Drug review, which describes users' experiences regarding a specific drug. Many Adverse Drug Reactions (ADRs) can be encountered. For example, "this medicine made me sleepy." These ADRs represent vital entities that should be extracted prior to the task of sentiment analysis in which the opinions of people are classified into positive or negative (Sohn *et al.*, 2011).

Most studies on the extraction of ADRs have relied on machine learning techniques involving models built on the basis of historical cases. Such models can extract new or unseen samples (Liu and Chen, 2015). However, the most significant factor of these techniques is a feature space that can be generated during model establishment. Features are descriptive characteristics that describe the occurrence of specific entities (Alshaikhdeeb and Ahmad, 2017; 2018). Discussing the feature space within the context

of extracting ADRs requires mentioning trigger terms, which are specific keywords that come before or after ADRs. Studies have utilized a set of trigger terms within the task of ADR extraction (Ebrahimi *et al.*, 2016; 2016). However, trigger terms still require various extensions because ADRs are numerous and have several synonyms and semantics.

This study aims to propose new trigger terms with various N-gram topologies, namely, unigram, bigram, trigram and quadgram. Two benchmark datasets are used within the experiments. Furthermore, two document representations have been utilized including Term Frequency Inverse Document Frequency (TFIDF) and Count Vector (CV). In addition, three classifiers, namely, Support Vector Machine (SVM), Naïve Bayes (NB) and Linear Regression (LR), are examined.

Related Works

Many studies have been proposed to extract adverse drug entities by using a wide range of features, along with machine learning techniques. For example, Yu (2016) used a set of features to identify drug-effect relation. The set of features contain Bag-

of-Words (BoW), where multiple numbers of topologies of N-gram, including unigram and bigram, have been used. Part-of-Speech (POS) tagging has been utilized to indicate the syntactic tags of terms. The WordNet lexicon has been used to indicate semantic correspondences. Four classifications, namely, decision tree, maximum entropy, NB and SVM, have been applied to identify drug–effect relation.

Mishra *et al.* (2015) utilized statistical features, such as term frequency and weighting terms, to identify drug-related entities from drug reviews. Along with statistical features, the Word Net lexicon is used to determine semantic relatedness. An SVM classifier has also been utilized to determine drug-related entities.

Pain *et al.* (2016) used the collected data from Twitter to provide an automatic drug–effect detection. They used a set of keywords and hashtags as trigger terms. The proposed features can identify numerous types of drug–effect entities by using an SVM classifier.

Ebrahimi *et al.* (2016) employed a set of medical concepts with specifically named entities as trigger terms to determine the side effects of drugs from medical reviews. POS tagging has also been utilized to identify the syntactic tag of terms. Two classifiers, namely, a rule-based classification method and SVM, have been adopted to detect the side effects of drugs.

Plachouras *et al.* (2016) utilized a set of trigger terms or Gazetteers features, along with an N-gram representation, to extract adverse drug events from Twitter reviews. They applied the SVM classification method to accommodate the final extraction based on the proposed features.

Moh *et al.* (2017) used a combination of morphological and semantic features to identify adverse drug events. Among the applied morphological features are negations and question marks. The semantic feature is utilized with SentiWordNet. SVM and NB have also been applied to perform a classification task.

Lee *et al.* (2017) have proposed a deep learning approach for extracting ADRs. The proposed approach utilized a semi-supervised Convolutional Neural Network (CNN) by giving it an unlabeled data brought from Twitter. The proposed approach showed superior performance compared to the traditional supervised ones.

Similarly, Cocos *et al.* (2017) have proposed a deep learning approach based on word embedding for extracting ADRs. The authors have brought vast amount of social information (particularly from Twitter) and train a Recurrent Neural Network (RNN) on such information to generate word embedding. The proposed method has been compared against the state of the art that were relying on lexicon-based approaches. Results showed an outperformance for the proposed method.

Wang *et al.* (2019) have proposed a deep neural network for extracting ADRs. The proposed method has utilized a pre-trained model for biomedical word embedding. Consequentially, the proposed method has been tested on a gold standard dataset. Results showed an outperformance for the proposed method against the baseline ones.

Materials and Methods

Materials: Dataset

The dataset used in the experiments is a set of drug reviews from different categories and from patients' comments about ADRs available on different discussion forums on health websites and social media in English language. Thus, two datasets are used:

Dataset 1: Ebrahimi *et al.* (2016) is annotated by a medical expert. A total of 225 drug reviews are randomly selected from www.drugratingz.com for manual annotation. These reviews are related to diverse categories, such as pain relief and antidepressant drugs. Indeed, the comment sections of drug reviews in this website are full of sentences containing drug side effects and the role of the algorithm is to identify these side effects correctly. A total of 70 reviews are to generate rules manually and 155 reviews are assigned as a test set

Dataset 2: The annotated ADR review dataset is used in Yates and Goharian (2013). The review dataset is collected from drug review social media sites, namely, askapatient.com, drugs.com and drugratingz.com

Table 1 shows the details of each dataset.

The proposed method consists of three main phases (Fig. 1). The first phase is related to the datasets used within the experiments, along with the required preprocessing tasks that are intended to turn the data into an appropriate form. The second phase is related to feature extraction, i.e., trigger terms, which represent the core of this study. Lastly, the third phase involves the application of machine learning techniques to classify ADRs based on the utilized features. Each phase is discussed in further detail in the next subsections.

Phase 1: Input and Preprocessing

Both datasets have to undergo the preprocessing task. The text segmentation stage aims to run some of the preprocessing algorithms on the corpus to prepare it for the next phases. The aforementioned tasks can be illustrated as follows:

1. Sentence splitting: This task aims to split a text into a series of sentences by identifying sentence boundaries. For this purpose, the Natural Language Tool Kit (NLTK) library is used to achieve this task
2. Tokenization: This task aims to split a text stream into a series of tokens. Similarly, the NLTK library is used
3. Stemming: This task aims to reduce inflected (or sometimes derived) words to their word stem, base, or root form of a particular set of words by removing various suffixes while preserving the meaning of the word
4. Stop word removal: This task aims to remove the frequent words of a language that does not carry any significant information on their own. These words are often removed at the preprocessing stage to reduce the number of less informative features known as noise data
5. POS tagging: This task aims to identify the words with their POS categories, such as nouns, verbs, adjectives and adverbs

Phase 2: Extended Trigger Terms

In this phase, a combination of lexical, syntactical and contextual expressions and trigger terms is used to detect the adverse side effects of drugs. Trigger terms are extracted from the state of art by analyzing the sentences containing ADR and trigger term. Two lists, namely, existing terms and new trigger terms, are created (Tables 2, 3 and 4).

Table 2 shows the trigger terms extracted by Ebrahimi *et al.* (2016). Among the terms, “caused” or “makes” are associated with ADRs in their dataset.

Table 3 shows the trigger terms in Yates and Goharian (2013) dataset, whose terms are similar to those in Ebrahimi *et al.* (2016) dataset. These terms (e.g., “caused” and “made me”) are also related to ADRs.

Building the Extended Trigger Terms

This study utilizes a statistical technique, namely, Point-wise Mutual Information (PMI), to identify new and extended trigger terms. This technique aims to examine the co-occurrence among terms. Both datasets are being annotated already. As such, PMI is applied to terms that frequently occur with ADRs. PMI can be computed on the basis of the following equation (Zhang *et al.*, 2009):

$$PMI(ADR, t_i) = \log \frac{P(ADR, t_i)}{P(ADR) \times P(t_i)} \quad (1)$$

where, $P(ADR)$ refers to the probability of individual ADR occurrence; $P(t_i)$ denotes the probability of the individual occurrence of certain terms and $P(ADR, t_i)$ corresponds to the co-occurrence among ADR and certain terms. The highest value of PMI indicates a high correlation among the two terms.

The results of PMI on both datasets reveal trigger terms that are similar to the baseline. Therefore, our study implements a manual filtering task to exclude the ones used by the baseline. With this filtering approach, new and extended trigger terms are identified. Table 4 shows a sample of these proposed extended terms associated with some example patterns from both datasets.

The sentences are represented in a feature vector that contains the selected features. Such a representation aims to articulate the distinctive terms in separated attributes. In this regard, every sentence is examined on the basis of the occurrence of such terms (i.e., whether or not a term occurs in a sentence). Here, the features are the distinctive terms and two frequency topologies, namely, Term Frequency –Inverse Document Frequency (TFIDF) and count vector, are depicted.

In count vector, the aim is to simulate the occurrence of terms as binary representation:

*“If a term occurs, it is represented as ‘1’;
 otherwise, it is represented as ‘0’”.*

TFIDF aims to represent the frequency of terms as real values that indicate the ratio of occurrence between a term and a sentence, along with a term with other sentences, which can be computed using the following equation (Chen *et al.*, 2016):

$$TFIDF = (t / d) = t_{fd} \times \log \frac{N}{N_t} \quad (2)$$

where, t_{fd} refers to the occurrence of the term in a particular document. The document in our study is a metaphor for a sentence. N is the number of the total documents (i.e., sentences) and N_t is the document that contain the term t .

Apart from frequency, the terms have been articulated in the N-gram representation by using four topologies, namely, unigram (i.e., one term), bigram (i.e., two terms), trigram (i.e., three terms) and quadgram (i.e., four terms).

Phase 3: Training Model

In this phase, machine learning is applied to classify ADRs. Classification methods, including SVM, NB and LR, are used to evaluate the performance based on f-measure.

The first classification method is SVM, which works by determining an accurate separator between data instances in a 2-dimensional space. Such a separator can be computed using the following equation (Ebrahimi, *et al.*, 2016):

$$f(\vec{x}) = \text{sgn}((\vec{x} \times \vec{w}) + b) = \begin{cases} +1: & (\vec{x} \times \vec{w}) + b > 0 \\ -1: & \text{Otherwise} \end{cases} \quad (3)$$

where, $d+$ ($d-$) denote the shortest path between the positive and negative examples.

NB is working by identifying the probabilities of classes for the data instances. Such a probability can be calculated using the following equation (Elhadad *et al.*, 2019):

$$P(C_i | d) = \frac{P(C_i)P(d | C_i)}{P(d)} \quad (4)$$

where, $P(C_i|d)$ is the posterior probability of class C_i given the predictor (x , attributes).

LR works by determining the linear equation of class probability, which can be depicted as follows (Montgomery, 2015):

$$y = a + bx, \quad (5)$$

where X is the dependent variable, a is the y -intercept and b is the slope of the line.

The evaluation involving f -measure can be depicted by the following equation:

$$F1 - score = 2 \times \frac{\frac{TF}{TF + FF} \times \frac{RF}{TF + FN}}{\frac{TF}{TF + FF} + \frac{RF}{TF + FN}} \quad (5)$$

where, TP is the correctly classified ADR , FP is the incorrectly classified ADR and FN is the correctly classified ADR in accordance with the total number of ADR s.

The three classifiers are trained on the extracted patterns produced by the proposed trigger terms and the benchmark ones. This training aims to build a model that can classify new data in the testing phase. During the training, the model of each classifier learns the cases of the potential occurrence of ADR s. Table 5 shows the experimental settings.

Table 4: Samples of extracted trigger terms

Comment_No	Sentence_No	Patterns	Trigger Terms	ADR Found
10	6	DT nn have caused (ADR)	have caused	Bad reaction
11	3	Prp suffer (ADR)	suffer	High blood
15	1	Jj side effect in(ADR)	side effect	arthritis
20	12	makes you (ADR)	makes you	Binge
24	3	Prp had never suffered from(ADR)	had never suffered	depression
26	2	Prp made me (ADR)	made me	Depressed
60	3	my side effect (ADR)	my side effect	stress

Table 5: Experimental settings

Experiment	Description
Feature	1. Baseline trigger terms with TFIDF (Unigram, Bigram, Trigram and Quadgram) 2. Baseline trigger terms with count vector (Unigram, Bigram, Trigram and Quadgram) 3. Proposed trigger terms with TFIDF (Unigram, Bigram, Trigram and Quadgram) 4. Proposed trigger terms with count vector (Unigram, Bigram, Trigram and Quadgram)
Classifiers	1. SVM 2. NB 3. LR
Dataset	1. Benchmark trigger terms dataset 1 (Ebrahimi, <i>et al.</i> , 2016) 2. Benchmark trigger terms dataset 2 (Yates and Goharian, 2013)
Training and Testing	70% for training and 30% for testing

Table 1: Dataset details

Attributes	Dataset 1 (Ebrahimi <i>et al.</i> , 2016)	Dataset 2 (Yates and Goharian, 2013)
Number of total reviews	225 (labelled 157)	2500 (labeled 246)
Number of sentences	1212	944
Number of ADR	372	982

Table 2: List of benchmark and proposed trigger terms based on dataset 1 (Ebrahimi, *et al.*, 2016)

Benchmark trigger terms (Ebrahimi <i>et al.</i> , 2016)	Proposed trigger terms
Caused	Causing effects of the drug
Causes	Getting off
Can cause	Wonder for
Caused an	Have been suffering
Cant cause	Have not got
Caused me	Short term effects
Makes you	Suffering with
Made me	Chronic
Make me	I had problems with
Makes people	Really helps

Table 3: List of benchmark and proposed trigger terms based on (Yates and Goharian, 2013)

Benchmark trigger terms (Yates and Goharian, 2013)	Proposed trigger terms
Caused	Started having
Causes	Began having
Makes you	Still have
Made me	Also have
Side effects	Had some
Side effect	Have some
Any side effects	Was having
I have	I am having
Get	Extreme
Side effect	Felt like

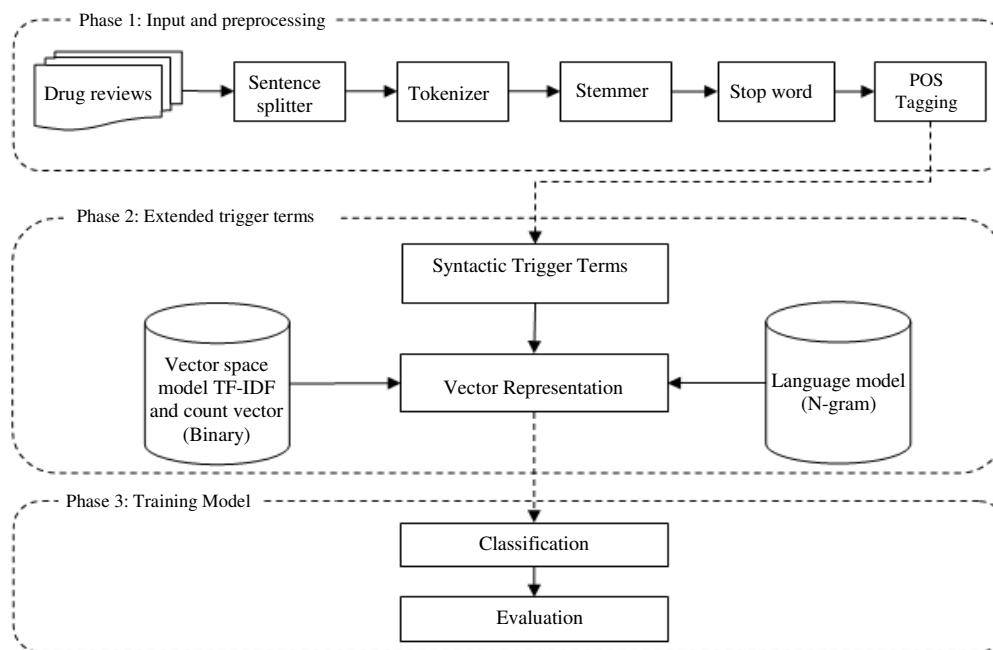


Fig. 1: Proposed extended trigger terms

Results

Multiple experiments have been conducted to examine different trigger terms (i.e., baseline vs. proposed), different representations (i.e., count vector vs. TFIDF) and multiple n-gram topologies (i.e., unigram, bigram, trigram and quadgram). In addition, multiple classifiers, including SVM, NB and LR, have been addressed and their corresponding results have been computed using the common information retrieval metric F1-score. Table 6 depicts the results of the first dataset.

In Tables 6 and 7, increasing the grams of terms affects the F1-score in which the quadgram shows the highest values among the topologies. In some cases, both trigram and quadgram have a similar performance (NB and LR for both experiments in Tables 6 and 7). This finding implies the usefulness of examining multigram terms.

The F1-score values of the count vector (Table 6) are higher than those of the TFIDF (Table 7). This condition is applied to all classifiers and both baseline and proposed ones. This result also implies the significance of using binary representation (i.e., count vector) rather than numeric representation (i.e., TFIDF).

However, the proposed and baseline results should be compared to validate the proposed trigger terms. Apparently, all the experiments show that the proposed method outperforms the baseline. In particular, the highest result is achieved by using the proposed trigger terms via the count vector with the SVM and the quadgram term. The result of F1-score is 88% (Table 6), demonstrating the effectiveness of the proposed trigger terms.

In Tables 8 and 9, similar to the result of dataset 1, the increase in the term gram affects the F1-score; that is, the quadgram results show the highest values. However, in some cases, trigram and quadgram have a similar performance (SVM and NB for both experiments in Tables 8 and 9). This result implies the effectiveness of the proposed approach in examining multigram terms.

Unlike the results of dataset 1, the F1-score values of the TFIDF (Table 9) are slightly higher than those of the count vector (Table 8). In terms of the comparison between the proposed and baseline results, all the experiments reveal that the proposed method outperforms the baseline. In particular, the highest results are achieved by the proposed trigger terms via the TFIDF with SVM and both trigram and quadgram terms and the result of F1-score is 69%. The result of dataset 1 is 88%, which is higher than that of dataset 2 (69%). This difference is attributed to the variance overlapping between the labels existing in the two datasets.

In general, the performance of the proposed trigger terms is superior to that of the baseline ones in terms of detecting ADRs. This finding implies the effectiveness of proposing extended trigger terms in terms of extracting ADRs.

Apart from the traditional baseline which utilized conventional approaches such as SVM, NB and others, it is necessary to compare the proposed method against recent methods that employed much sophisticated techniques. In fact, Lee *et al.* (2017) have used a deep learning approach of CNN to extract ADRs and acquired an F1-score of 64.5%. Comparing such results against the obtained ones by the proposed method reveals that the proposed method is still competitive.

Table 8: Count vector results of dataset 2 (Yates and Goharian, 2013)

N-gram	Count vector (F1-score)					
	SVM		NB		LR	
	Baseline	Proposed	Baseline	Proposed	Baseline	Proposed
Unigram	0.54	0.55	0.38	0.42	0.55	0.57
Bigram	0.59	0.65	0.51	0.57	0.59	0.64
Trigram	0.59	<u>0.67</u>	0.52	<u>0.61</u>	0.61	0.66
Quadgram	0.56	<u>0.67</u>	0.52	<u>0.61</u>	0.61	<u>0.67</u>

Table 9: TFIDF results of dataset 2 (Yates and Goharian, 2013)

N-gram	TFIDF (F1-score)					
	SVM		NB		LR	
	Baseline	Proposed	Baseline	Proposed	Baseline	Proposed
Unigram	0.54	0.54	0.38	0.44	0.54	0.54
Bigram	0.58	0.67	0.51	0.58	0.59	0.66
Trigram	0.60	<u>0.69</u>	0.54	<u>0.61</u>	0.60	<u>0.68</u>
Quadgram	0.60	<u>0.69</u>	0.54	<u>0.61</u>	0.60	<u>0.68</u>

However, other studies such as Cocos *et al.* (2017) and Wang *et al.* (2019) whom utilized much sophisticated deep learning approaches, have obtained an F1-score higher the proposed method as 75.5% and 84.4% respectively. Yet, their approaches were requiring a pre-trained data of embedding for the medical words. Considering the feature engineering that has been utilized by the proposed method, it is clear that the proposed method is still considered to be less complicated.

Conclusion

This study proposed an extended set of trigger terms for detecting ADRs. These trigger terms were compared with the baseline ones by using two benchmark datasets. Experiments involved three classifiers, namely, SVM, NB and LR and multiple N-gram topologies, including unigram, bigram, trigram and quadgram. The proposed trigger terms achieved higher results than the baseline ones when quadgram and SVM classification were used. Further studies on feature types would facilitate the process of detecting ADRs.

Acknowledgment

This work was supported by Universiti Kebangsaan Malaysia (Grant code: GGPM-2017-025).

Author's Contributions

Rami Naim Mohammad Yousef: selected the topic and analyzed the literature along with proposing and implementing the methods.

Sabrina Tiun and Nazlia Omar: supervising the research by reviewing the manuscript iteratively and giving suggestions and recommendations.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

- Alshaikhdeeb, B. and K. Ahmad, 2017. Feature selection for chemical compound extraction using wrapper approach with Naive Bayes classifier. Proceedings of the 6th International Conference on Electrical Engineering and Informatics, Nov. 25-27, IEEE Xplore press, Langkawi, Malaysia, pp: 1-6.
 DOI: 10.1109/ICEEI.2017.8312421
- Alshaikhdeeb, B. and K. Ahmad, 2018. Comparative analysis of different data representations for the task of chemical compound extraction. Int. J. Advanced Sci. Eng. Inform. Technol., 8: 2189-2195.
 DOI: 10.18517/ijaseit.8.5.6432
- Chen, K., Z. Zuping L. Jun and Z. Hao, 2016. Turning from TF-IDF to TF-IGM for term weighting in text classification. Expert Syst. Applic., 66: 245-260.
 DOI: 10.1016/j.eswa.2016.09.009
- Cocos, A., A.G. Fiks and A.J. Masino, 2017. Deep learning for pharmacovigilance: Recurrent neural network architectures for labeling adverse drug reactions in twitter posts. J. Am. Med. Informatics Assoc, 24: 813-821. DOI: 10.1093/JAMIA/OCW180
- Denecke, K. and Y. Deng, 2015. Sentiment analysis in medical settings: New opportunities and challenges. Artificial Intel. Med., 64: 17-27.
 DOI: 10.1016/j.artmed.2015.03.006
- Ebrahimi, M., A.H. Yazdavar, N. Salim and S. Eltyeb, 2016. Recognition of side effects as implicit-opinion words in drug reviews. Online Inform. Rev., 40: 1018-1032.
 DOI: 10.1108/OIR-06-2015-0208

- Ebrahimi, M., A.H. Yazdavar, N.B. Salim, S. Noekhah and D.L. Paris, 2016. Drug side effect detection as implicit opinion from medical reviews. *Res. J. Applied Sci. Eng. Technol.*, 12: 1086-1094. DOI: 10.19026/rjaset.12.2850
- Elhadad, M.K., K.F. Li and F. Gebali, 2019. Sentiment Analysis of Arabic and English Tweets. In: *Web, Artificial Intelligence and Network Applications*, Springer, Cham, ISBN-10: 978-3-030-15035-8, pp: 334-348.
- Lee, K., A. Qadir, S.A. Hasan, V. Datla and A. Prakash *et al.* 2017. Adverse drug event detection in tweets with semi-supervised convolutional neural networks. *Proceedings of the 26th International Conference on World Wide Web, Conferences Steering Committee*, Apr. 03-07, ACM, Perth, Australia, pp: 705-714. DOI: 10.1145/3038912.3052671
- Liu, A. and M. Chen. 2015. A research framework for pharmacovigilance in health social media: Identification and evaluation of patient adverse drug event reports. *J. Biomed. Informatics*, 58: 268-279.
- Mishra, A., A. Malviya and S. Aggarwal, 2015. Towards automatic pharmacovigilance: Analysing patient reviews and sentiment on oncological drugs. *Proceedings of the IEEE International Conference on Data Mining Workshop*, Nov. 14-17, IEEE Xplore Press, Atlantic City, NJ, USA, pp: 1402-1409. DOI: 10.1109/ICDMW.2015.230
- Moh, M., T.S., Moh and Y. Peng, 2017. On adverse drug event extractions using twitter sentiment analysis. *Netw. Model. Analysis Health Informatics Bio.*, 6: 18-18. DOI: 10.1007/s13721-017-0159-4
- Montgomery, A., 2015. *Introduction to Linear Regression Analysis*. 1st Edn., John Wiley and Sons.
- Pain, J., L. Jessie, Q. Adam and B. Anja, 2016. Analysis of twitter data for postmarketing surveillance in pharmacovigilance, *Proceedings of the 2nd Workshop on Noisy User-Generated Text*, Dec. 11, Osaka, Japan, pp: 6-39.
- Plachouras, V., J.L. Leidner and A.G. Garrow, 2016. Quantifying self-reported adverse drug events on twitter: Signal and topic analysis. *Proceedings of the 7th International Conference on Social Media and Society*, Jul. 11-13, London, United Kingdom, ACM, pp: 1-6. DOI: 10.1145/2930971.2930977
- Sohn, S., J.P. Kocher, C.G. Chute and G.K. Savova, 2011. Drug side effect extraction from clinical narratives of psychiatry and psychology patients. *J. Am. Med. Informatics Assoc.*, 18: i144-i149. DOI: 10.1136/amiajnl-2011-000351
- Wang, C.S., P.J. Li, C.L. Cheng, S.H. Tai and Yea, H.K. Yang *et al.* 2019. Detecting potential adverse drug reactions using a deep neural network model. *J. Med. Internet Res.*, 21: e11016- e11016.
- Yates, A. and N. Goharian, 2013. ADRTrace: Detecting expected and unexpected adverse drug reactions from user reviews on social media sites. In: *Advances in Information Retrieval*, Serdyukov, P. (Ed.), Springer, Berlin, Heidelberg, ISBN-10: 978-3-642-36972-8, pp: 816-819.
- Yu, A., 2016. Towards extracting drug-effect relation from Twitter: A supervised learning approach. *Proceedings of the IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing and IEEE International Conference on Intelligent Data and Security, (IDS' 16)*, IEEE, pp: 339-344.
- Zhang, W., T. Yoshida, X. Tang and T.B. Ho, 2009. Improving effectiveness of mutual information for substantival multiword expression extraction. *Expert Syst. Appl.*, 36: 10919-10930. DOI: 10.1016/j.eswa.2009.02.026