ICIC Express Letters
Volume **15**, Number **8**, August **2021**

ICIC International ©2021 ISSN 1881-803X
pp. **907–914**

EXTRACTING ADVERSE DRUG REACTION USING LATENT SEMANTIC ANALYSIS FROM MEDICAL SOCIAL MEDIA REVIEWS

Alaa Hussein Abed[1,*], Sanaa Ali Jabber[2]
and Ammar Abdul-Jabbar Altameemi[3]

[1]Department of Clinical Laboratory Sciences
College of Pharmacy
[3]Biomedical Engineering Department
College of Engineering
University of Kerbala
Hilla Road, Freha, Kerbala 56001, Iraq
*Corresponding author: alaa.h@uokerbala.edu.iq

[2]Faculty of Administration and Economics
Al-Muthanna University
Baghdad Road, Samawah, Al-Muthanna 66001, Iraq

Received November 2020; accepted February 2021

Abstract. *Adverse Drug Reaction (ADR) extraction is the process of detecting mentions of unpleasant reactions resulted from taking medicine within reviews of drug websites. Several research studies have addressed the process of detecting ADR from social medical text through machine learning or rule-based techniques. The majority of such studies have concentrated on manually curated keywords that frequently occurred with the ADRs; these keywords are known as trigger terms, yet it is difficult to identify these terms since reviews might include non-formal terms. Therefore, this paper aims at proposing a statistical method of Latent Semantic Analysis (LSA) in which the contextual occurrence of ADRs can be identified. To do so, different term representations have been utilized including Bag of Words (BoW), Count Vector (CV), and Term Frequency Inverse Document Frequency (TFIDF). Consequentially, three machine learning techniques including Naïve Bayes (NB), Logistics Regression (LR), and Support Vector Machine (SVM) are being used to detect the ADRs based on the feature matrix produced by LSA. Using a benchmark dataset for medical social reviews, the proposed LSA has shown a remarkable enhancement compared to the state of the art. This has been demonstrated where the highest F1-score result was obtained by the LR classifier with LSA and got 82%. This can demonstrate the efficacy of using LSA instead of manually curated features for ADR detection.*

Keywords: Medical social reviews, Adverse drug reaction extraction, Latent semantic analysis, Naïve Bayes, Support vector machine

1. **Introduction.** The dramatic expansion of Web 2.0 has enabled diverse industries to conduct business over the Internet [1]. Plenty of retailers, stores, and factories have become much more interested in developing their online shopping websites where users can search, examine, determine details and place an order of a particular product.

Within such a business platform, a valuable section has become very fundamental for any online shopping store which is the review comments. In this section, users who purchased the product can freely express their experience where they can describe their feedback in terms of the quality of the product, defects, or any obstacles that might be happened during their purchasing. In this regard, other consumers would considerably benefit from such valuable comments. The comments or reviews of a particular product have caught both the research and industrial sectors' attention due to the wide range of

DOI: 10.24507/icicel.15.08.907

opportunities that can be achieved from analyzing people's opinions. One of the domains that have been studied in terms of social reviews is the medical domain. This specific domain would have more interesting opportunities that are not only related to consumer satisfaction but also identifying other challenging factors [2]. Hence, many research studies aimed at collecting reviews from online medical shopping stores to tag the ADRs within the people's comments. This has been conducted to train the machine to automatically identify any potential ADR. However, the literature has extensively invested in the use of trigger terms for both paradigms. These terms are informative keywords that have significant correlations with the ADR terms. For example, the phrase "I have severe pain" would have an ADR of "pain" and a keyword of "severe". Organizing a list of trigger terms would have a considerable impact on extracting ADR. However, it requires tremendous linguistic analysis such as the use of external knowledge sources (e.g., dictionaries, and lexicons) or the use of syntactic tools such as Part-of-Speech (POS) tagging or parser. Besides, utilizing trigger terms might mislead the classifier when they appear without the occurrence of ADR (e.g., severe pain led me to this medicine). Therefore, this study aims to substitute the use of trigger terms and replace them with a much more efficient technique that might be able to determine the semantic analysis of ADR occurrence. This technique is Latent Semantic Analysis (LSA) which is intended to analyze the semantic of particular text statistically and without the use of any external knowledge source. Based on the analysis of LSA, three machine learning algorithms will be trained to extract ADR including NB, LR, and SVM. The results of extracting ADR using the proposed LSA will be compared with the state-of-the-art trigger terms based technique.

The paper is organized as follows. Section 2 highlights the related work and their enduring limitations. Section 3 discusses the proposed method and its dependencies. Section 4 shows the results and discussion. Finally, Section 5 concludes the paper.

2. **Related Work.** To clarify the problem behind extracting ADRs within social reviews, it is necessary to highlight the state-of-the-art techniques used in the literature. The earliest efforts were depending on the Bag-of-Words (BoW) approach in which the terms within the reviews are being represented in the feature space. For instance, Mishra et al. [3] have used the BoW with some statistics such as term frequency to extract drug-related entities from social reviews. The authors have trained some machine learning algorithms such as SVM on such representation with the help of the WordNet dictionary to determine semantic correspondences. Similarly, Yu et al. [4] used BoW with multiple N-gram representation such as unigram and bigram to extract drug effect relations. The authors have utilized also some machine learning algorithms such as SVM and NB for training on such a BoW representation.

Afterward, some studies have attempted to enhance the BoW representation by selecting significant terms such as the trigger terms. For instance, Pain et al. [5] have used trigger terms based on Twitter hashtags to identify drug-effect within tweets. The authors have utilized the SVM classifier to train on such a representation. Similarly, Ebrahimi et al. [6] proposed a new set of trigger terms using the Unified Medical Language System (UMLS) for detecting side effects from medical social reviews. The authors have utilized two detection techniques including rule-based and SVM. In the same regard, Plachouras et al. [7] extracted trigger terms using Gazetteers to detect adverse drug events from tweets. The authors have also utilized the SVM classifier. Besides, Moh et al. [8] used the lexicon of SentiWordNet to get additional trigger terms for detecting ADR from tweets. The authors have utilized both NB and SVM classifiers.

Consequentially, further researches are depicted to develop, extend, and expand the trigger terms range. For instance, Kiritchenko et al. [9] have expanded the trigger terms with more domain-specific keywords to detect ADRs throughout tweets. Wu et al. [10] expanded the trigger terms based on utilized abbreviations to detect drug mentions within tweets. Yousef et al. [11] expanded the trigger terms using a syntactic tool of POS tagging and Pointwise Mutual Information (PMI) to detect ADR within medical social reviews. PMI has been used to identify the most frequent syntactic patterns that occurred along with ADR. Such frequent patterns facilitated the process of expanding the trigger terms. Lastly, the authors have utilized three classifiers including NB, LR, and SVM. Finally, Zhang et al. [12] examined the role of semantic source and syntactic POS tagging to generate pairs of trigger terms for detecting ADRs from social reviews.

After highlighting the techniques used in the literature to detect ADRs, it is obvious that the main dependency was depicted by utilizing trigger terms, keywords, and other semantics that require the use of external knowledge sources. The main limitation behind such an approach lies in the demand for building a source for such keywords. Even after building such a source, it is sometimes quite misinformative where those terms might not occur with ADRs. Therefore, this study aims at utilizing a statistical technique that does not require the use of external knowledge sources or any pre-defined semantic list.

3. **Proposed LSA.** The framework of the proposed method consists of multiple components as shown in Figure 1. First, the dataset of medical reviews is depicted where the drug review documents are being used for further analysis. Then, multiple cleaning tasks will be applied such as tokenization, stopwords removal, and stemming. Such cleaning tasks aim at representing the text in an appropriate form for processing. Consequentially, the BoW representation will be used to capture the cleaning terms along with the review documents. After that, a frequency analysis is depicted where the Count Vector (CV) and Term Frequency Inverse Document Frequency (TFIDF) are used to compute the occurrences of each term per every review document. Hence, the proposed LSA will be applied to such a TFIDF matrix where the Singular Value Decomposition (SVD) is being conducted to determine semantic approximation. Lastly, three classifiers of NB, LR, and SVM will be trained and tested on the resulted matrix of SVD. The following subsections will tackle each component independently.

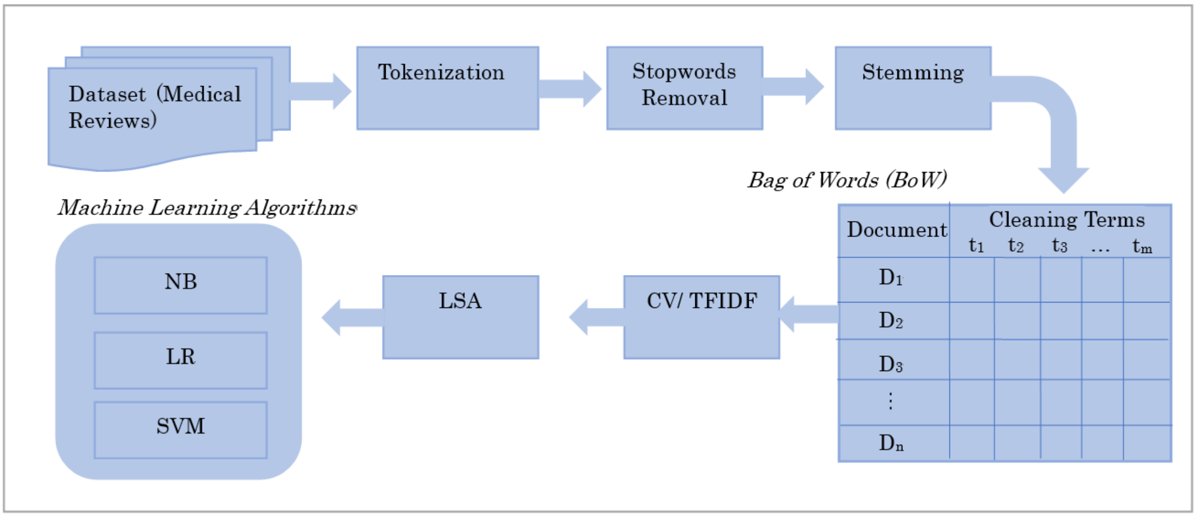

FIGURE 1. The framework of the proposed method

3.1. **Medical reviews dataset.** In this section, the dataset that will be examined in the experiments is tackled. It is a benchmark dataset introduced by Yates and Goharian [13] which has been collected from the comment section of drug websites including askapatient.com, drugs.com, and drugratingz.com. It contains separated reviews along with their class label as shown in Table 1. The class label is either 1 or 0 where 1 refers to the existence of ADR and 0 refers to the absence of ADR.

Table 1. Sample reviews of the dataset

Review documents	Text	Class label
D_1	My joint pain is very severe	1
D_2	Lower back pain	1
D_3	Switched by oncologist to Aromasin	0

As shown in Table 1, the first two reviews contained ADRs depicted by the word ‘pain’. However, the last review is just an objective review where the consumer described a fact without mentioning any side effects.

3.2. **Cleaning tasks.** In this section, the main tasks of cleaning the text will be determined. The first task is tokenization where the aim is to separate each term within every review in order to enable further processing on the word-level. The second task is the stopword removal where insignificant terms such as ‘is’, ‘the’, and ‘of’ will be discarded. The third task is the stemming where the additional affixes such as ‘ing’, ‘ed’, and ‘er’ are removed. Table 2 shows a demonstration of applying the three cleaning tasks on the example review text.

Table 2. Applying cleaning tasks

Original review text	After tokenization	After stopwords removal	After stemming
My joint pain is very severe	[[My], [joint], [pain], [is], [very], [severe]]	[[joint], [pain], [severe]]	[[joint], [pain], [**sever**]]
Lower back pain	[[Lower], [back], [pain]]	[[Lower], [back], [pain]]	[[Lower], [back], [pain]]
Switched by oncologist to Aromasin	[[Switched], [by], [oncologist], [to], [Aromasin]]	[[Switched], [oncologist], [Aromasin]]	[[**Switch**], [oncologist], [Aromasin]]

After applying the cleaning tasks, the BoW representation will be used to depict each cleaned term within the feature space per each review document as shown in Table 3. In BoW, 1 indicates that the term occurs in the particular document, while 0 indicates the absence.

Table 3. BoW representation

Documents	Cleaned terms							
	joint	pain	sever	lower	back	switch	oncologist	Aromasin
D_1	1	1	1	0	0	0	0	0
D_2	0	1	0	1	1	0	0	0
D_3	0	0	0	0	0	1	1	1

3.3. **CV/TFIDF.** After obtaining the BoW representation, it is important to analyze each term statistically in which the frequency of terms will be computed separately following each review document. This task is essential to determine the significance of every term which will facilitate further analysis when using the proposed LSA. For this purpose, the CV will be used to calculate the frequency of each term which can be computed as in the following equation:

$$Count\ Vector\ (CV) = TF_{td} \tag{1}$$

where TF_{td} is the number of times the term t occurs in document d. Besides the individual frequency of each term, it is necessary to examine the frequency ratio for each term following every single document. For this purpose, the TFIDF analysis is used where it is calculated as in the following equation:

$$TFIDF_{(t\backslash d)} = TF_{td} \times IDF \tag{2}$$

where TF_{td} is the number of times the term t occurs in document d (i.e., CV), and IDF is the ratio between term frequency and the total number of documents [14]. IDF can be computed as follows:

$$IDF = \log(N/N_t) \tag{3}$$

where N is the total number of documents, and N_t is the number of documents contained the term t.

3.4. **LSA.** LSA is one of the statistical techniques that can determine the semantic correspondences among text without the need of utilizing any external knowledge source [15]. It works by identifying similar contexts which indeed would yield similar words in meaning, for example, the two phrases "this medicine makes me sleepy" and "this drug gets me sleepy" have similar context but with different words that have a similar meaning (e.g., medicine and drug, makes and gets). The key success of LSA lies in the ability to obtain large text in a specific domain where various semantic matches can be identified. Besides, LSA relies on the BoW matrix where the terms are being arranged following the documents. Hence, either CV or TFIDF can populate the BoW matrix's values. Since the work will be based on matrices, LSA utilizes SVD to factorize the input matrix and its sub-matrices. SVD can be computed based on the following equation [16]:

$$M = U\Sigma V^* \tag{4}$$

in which M is BoW matrix $n \times m$, U is the left singular matrix $m \times k$ where k is an integer number $k \in n$ and it refers to the number of topics T, Σ is a diagonal matrix $k \times k$, and V^* is the right singular matrix $k \times n$. To understand the matrices of SVD, Figure 2 represents them under our problem. As shown in Figure 2, there two singular matrices where the first refers to the term assignment to the generated topics, and the second refers to the distribution of topics over the review documents. LSA will generate a random number k of topics T which must be less than the number of documents n. Lastly, the diagonal matrix refers to the topic of importance where all topics will be compared to each other to merge similar topics. In other words, this matrix will filter the topics to generate a more accurate number of topics.

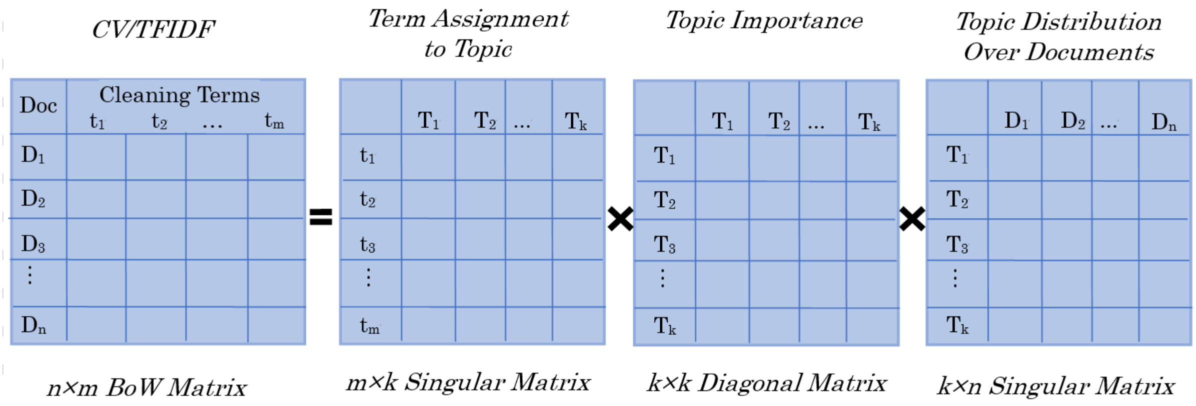

FIGURE 2. SVD matrices

3.5. **Machine learning algorithms.** After getting the resulted matrix of LSA, this section will take a place in which three classifiers of NB, LR, and SVM are being trained on 70% of the data and tested on the remaining 30% similar to the study of Yousef et al. [11]. The first classifier NB works by examining the probabilities of each class label over the review documents based on an independent feature probability. This probability is computed as follows [17,18].

$$P(C_i|d) = P(C_i)P(d|C_i)/P(d) \tag{5}$$

where $P(C_i|d)$ is the posterior probability of class C_i given predictor (x, attributes). Similar to NB, LR works by examining probabilities yet, it focuses on dependent feature probability using the linear equation of class probability that is calculated as follows [19]:

$$y = a + bx \tag{6}$$

where x articulates the dependent variables, a is the y-intercept and b represents the slope of the line. Lastly, the SVM classifier works by identifying a hyperplane which is a separator among different classes in the two-dimensional representation of the vector features. The hyperplane is calculated as follows [20-22]:

$$f(x) = sgn((x \times w) + b) = \begin{cases} +1 & ((x \times w) + b) > 0 \\ -1 & \text{Otherwise} \end{cases} \tag{7}$$

where x articulates the dependent variables, w is the distance between two nearest points of two different classes, and b is the distance between each point from the two classes and the hyperplane. To evaluate each classifier, the conventional metric of the F1-score will be used. Such a metric is the harmony between precision and recall. Precision focuses on the review documents that do not contain ADRs yet, it has been classified as containing ADR. Precision can be calculated as

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{8}$$

where TP refers to the number of correctly classified documents, and FP refers to the number of incorrectly non-ADR documents that are classified as ADR. On the other hand, recall focuses on the documents that contain ADRs but are classified as non-ADR containing documents. It can be computed as

$$\text{Recall} = \text{TP}/(\text{TP} + \text{TN}) \tag{9}$$

where TN refers to the number of incorrectly ADR documents that are classified as non-ADR. Hence, F1-score can be computed as follows:

$$\text{F1-score} = (2 \times \text{Precision} \times \text{Recall})/(\text{Precision} + \text{Recall}) \tag{10}$$

4. **Results.** In this section, the results of applying the proposed LSA along with the three machine learning algorithms will be depicted. This can be demonstrated twice: first using the CV matrix, and second using the TFIDF matrix. Besides the results of the proposed method, a baseline study of Yousef et al. [11] will be used for the comparison. Such a baseline study examined the same dataset used in this study which will facilitate the comparison. Table 4 shows the results of the F1-score for each experiment.

Table 4. Results of classification using F1-score

Classifiers	CV		TFIDF	
	Baseline [11]	Proposed	Baseline [11]	Proposed
NB	61%	71%	61%	71%
LR	67%	**82%**	68%	78%
SVM	67%	81%	69%	79%

As shown in Table 4, the results of the three classifiers have been remarkably enhanced when using the proposed LSA compared to the traditional trigger terms approach used by the baseline study for both CV and TFIDF. This has been depicted by CV where NB got 71% of F1-score with LSA compared to 61% using trigger terms. Besides, LR obtained 82% with LSA compared to 67% with trigger terms. Lastly, SVM gained 81% with LSA compared to 67% by the trigger terms. On the other hand, the results of TFIDF showed that NB got 71% with LSA compared to 61% with trigger terms. Whereas, LR obtained

78% with LSA compared to 68% with the trigger terms approach. Finally, SVM got 79% with LSA compared to 69% with the trigger terms. Obviously, the proposed LSA has remarkably improved the results of classifiers. The reason behind such outperformance lies in the ability to capture semantic correspondences which helps the identification of ADRs. In contrast, the trigger terms approach used by the baseline focuses on populating the feature space with keyword terms that might occur along with ADRs. Apparently, these keywords would not occur with ADR always in which capturing them in the feature space without addressing the semantic aspect would not help the identification of ADRs. Apart from the comparison between the proposed LSA and the traditional approach of trigger terms, it is noticeable that the results of applying LSA with CV were better than when applying LSA with TFIDF. This was expected from the study of Baroni et al. [23] who claimed that corpus-based techniques such as LSA would optimally benefit from CV rather than TFIDF due to the integer representation depicted in the former paradigm compared to the continuous representation in the latter paradigm. In the same regard, LR obtained the highest F1-score value of 82% via LSA with CV for the same reason where LR can optimally benefit from the integer representation of CV rather than the continuous representation of TFIDF.

5. **Conclusions.** This study presented the use of LSA for the purpose of ADR extraction from social drug reviews. Using a benchmark dataset, the proposed method showed not only a substitution way that does not need external knowledge source but also better performance of extraction compared to the traditional techniques. For future researches, the utilization of modern vector representation of word embedding would improve the detection performance.

REFERENCES

[1] B. Alshaikhdeeb and K. Ahmad, Integrating correlation clustering and agglomerative hierarchical clustering for holistic schema matching, *Journal of Computer Science*, vol.11, no.3, pp.484-489, 2015.

[2] B. Audeh, F. Bellet, M.-N. Beyens, A. Lillo-Le Louët and C. Bousquet, Use of social media for pharmacovigilance activities: Key findings and recommendations from the Vigi4Med project, *Drug Safety*, vol.43, no.9, pp.835-851, 2020.

[3] A. Mishra, A. Malviya and S. Aggarwal, Towards automatic pharmacovigilance: Analysing patient reviews and sentiment on oncological drugs, *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, pp.1402-1409, 2015.

[4] F. Yu, M. Moh and T.-S. Moh, Towards extracting drug-effect relation from Twitter: A supervised learning approach, *2016 IEEE the 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS)*, pp.339-344, 2016.

[5] J. Pain, J. Levacher, A. Quinqunel and A. Belz, Analysis of Twitter data for postmarketing surveillance in pharmacovigilance, *Proc. of the 2nd Workshop on Noisy User-Generated Text*, Osaka, Japan, pp.94-101, 2016.

[6] M. Ebrahimi, A. H. Yazdavar, N. Salim and S. Eltyeb, Recognition of side effects as implicit-opinion words in drug reviews, *Online Information Review*, vol.40, no.7, pp.1018-1032, 2016.

[7] V. Plachouras, J. L. Leidner and A. G. Garrow, Quantifying self-reported adverse drug events on Twitter: Signal and topic analysis, *Proc. of the 7th 2016 International Conference on Social Media & Society*, 2016.

[8] M. Moh, T.-S. Moh, Y. Peng and L. Wu, On adverse drug event extractions using Twitter sentiment analysis, *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol.6, no.1, p.18, 2017.

[9] S. Kiritchenko, S. M. Mohammad, J. Morin and B. de Bruijn, NRC-Canada at SMM4H shared task: Classifying Tweets mentioning adverse drug reactions and medication intake, *arXiv Preprint*, arXiv: 1805.04558, 2018.

[10] C. Wu, F. Wu, Z. Yuan, J. Liu, Y. Huang and X. Xie, MSA: Jointly detecting drug name and adverse drug reaction mentioning Tweets with multi-head self-attention, *Proc. of the 12th ACM International Conference on Web Search and Data*, pp.33-41, 2019.

[11] R. N. Yousef, S. Tiun and N. Omar, Extended trigger terms for extracting adverse drug reactions in social media texts, *Journal of Computer Science*, vol.15, no.6, pp.873-879, 2019.

[12] Y. Zhang, S. Cui and H. Gao, Adverse drug reaction detection on social media with deep linguistic features, *Journal of Biomedical Informatics*, vol.106, 2020.

[13] A. Yates and N. Goharian, ADRTrace: Detecting expected and unexpected adverse drug reactions from user reviews on social media sites, in *Advances in Information Retrieval. ECIR 2013. Lecture Notes in Computer Science*, P. Serdyukov et al. (eds.), Berlin, Heidelberg, Springer, 2013.

[14] K. Chen, Z. Zhang, J. Long and H. Zhang, Turning from TF-IDF to TF-IGM for term weighting in text classification, *Expert Systems with Applications*, vol.66, pp.245-260, 2016.

[15] S. R. Vrana, R. S. Bono, A. Konig and G. C. Scalzo, Assessing the coherence of narratives of traumatic events with latent semantic analysis, *Psychological Trauma: Theory, Research, Practice, and Policy*, vol.11, no.5, pp.521-524, 2019.

[16] L. Cagliero, P. Garza and E. Baralis, ELSA: A multilingual document summarization algorithm based on frequent itemsets and latent semantic analysis, *ACM Transactions on Information Systems (TOIS)*, vol.37, no.2, pp.1-33, 2019.

[17] B. Alshaikhdeeb and K. Ahmad, Comparative analysis of different data representations for the task of chemical compound extraction, *International Journal on Advanced Science, Engineering and Information Technology*, vol.8, no.5, pp.2189-2195, 2018.

[18] B. Alshaikhdeeb and K. Ahmad, Feature selection for chemical compound extraction using wrapper approach with Naive Bayes classifier, *2017 the 6th International Conference on Electrical Engineering and Informatics (ICEEI)*, pp.1-6, 2017.

[19] T. Rymarczyk, E. Kozłowski, G. Kłosowski and K. Niderla, Logistic regression for machine learning in process tomography, *Sensors*, vol.19, no.15, 2019.

[20] C. Meng, S. Jin, L. Wang, F. Guo and Q. Zou, AOPs-SVM: A sequence-based classifier of antioxidant proteins using a support vector machine, *Frontiers in Bioengineering and Biotechnology*, vol.7, DOI: 10.3389/fbioe.2019.00224, 2019.

[21] T.-K. Lin, An edge-feature-description-based scheme combined with support vector machines for the detection of vortex-induced vibration, *International Journal of Innovative Computing, Information and Control*, vol.14, no.3, pp.833-845, 2018.

[22] A. M. Farayola, A. N. Hasan and A. Ali, Efficient photovoltaic MPPT system using coarse Gaussian support vector machine and artificial neural network techniques, *International Journal of Innovative Computing, Information and Control*, vol.14, no.1, pp.323-339, 2018.

[23] M. Baroni, G. Dinu and G. Kruszewski, Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors, *Proc. of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp.238-247, 2014.