International Journal of Data Science and Advanced Analytics (ISSN: 2563-4429)

Using Supervised Machine Learning Models and Natural Language

Sidarth Mohan[1], Jolnar Assi[2] and Ammar H Mohammed[3]
[1, 2, 3]Liverpool John Moores University, UK, Traders Island Ltd, UK, Iraqi Prime Minister's Office, Iraq
[1]sidharthmmec@gmail.com, [2]jaa06@hotmail.com, [3]ammarhussen659@gmail.com

Corresponding author email: sidharthmmec@gmail.com

Abstract— **Social media has gained popularity over the last decade due to its ease of access and providing large amount of information to people. In seconds, users are able to access information from social media related to politics, life-style, science and money other fields. However, data obtained from social media platforms represent a mixture of fake and real news. Fake news are in-tended to deceive people and change their attitudes and beliefs. Machine learning algorithms have shown successful in classifying real from fake news. Nonetheless when applying machine learning models in this context related to limitations in the dataset type, balance or skewness. Hence, data pre-processing is essential prior to application of machine learning models. Therefore, this work evaluated the use of supervised machine learning models with different data pre-processing approaches for classification of fake news obtained from social media platforms. Different pre-processing techniques have been applied related to feature extraction and feature selection alongside four machine learning models being logistic regression, decision trees, random forest and extreme gradient boost. The findings showed that random forest and extreme gradient boost with bi-gram feature extraction and chi-squared feature selection showed the best performance. Future work involves using the proposed model to detecting fake news in different con-text and different languages.**

Keywords—**Social Media; Fake News; Machine Learning; Feature Selection; Feature Extraction**

1. Introduction

The use of social media has surged over the last decade for many purposes related to connecting with families/friends, shopping, entertainment, news or looking for jobs [1]. Social media platforms offer enormous amounts of data for users in seconds and that contribution to their increased popularity [2]. Yet the authenticity of data over social media platform has always been a question, where these platforms often contain a mixture of real and fake news [3]. This is because there is no control over who posts over social media platforms that in turn allow any user to post any content. Yet the ease of use of these platforms and their instant feedback have contributed to their increased popularity.

Fake news are defined as incorrect information spread in an unsuitable form in or-der to deceive people by misleading people's beliefs and attitudes [4]. Fake news spread at a more rapid pace than real news because they are advertised in a more attractive and appealing way to people [5]. Moreover, fake news have hidden political agenda and have sensitive content the incite strong emotions from users (e.g. sympathy, outrage, anger) [6]. Social media bots help with spreading fake news quickly and without control [7].

This urges the need to distinguish between and real news and thereby authenticate information for people. Subsequently, several fact checking tools were created to warn consumers against fake news e.g. the International Fact Checking Network [8]. Yet these tools do manual checking and thus often fail to eliminate fake news that spread at an extremely rapid pace and in large amounts [3].

Subsequently, machine learning and deep learning algorithms (MLAs and DLAs) have been deployed for automatic detection of fake news [9-24]. MLAs have provided good accuracy in classifying real from fake news; yet the quality of the model depends on the quality of the dataset. Reported models had variable accuracy between 40 – 99% and included decision trees (DT) [9, 10], logistic regression (LR) [10, 11], random forest (RF) [9, 11-13], support vector machine (SVM) [10, 11], logistic regression (LR) [10, 11], deep neural networks (such as convolutional neural networks and recurrent neural networks) [14-23], gradient boost (e.g. adaptive gradient boost and extreme gradient boost) [12]. Yet datasets used for fake news detection with MLAs are often of small sample size, imbalanced or skewed [22, 23]. This in turn could give rise to overfitting of the model or poor prediction where the model could be biased towards one class of data.

Consequently, the present study built on the findings of previous studies and utilised MLAs for classifying real from fake news from social media platforms using MLAs. The approach looks in detail into data pre-processing (including cleaning) approaches, feature extraction/selection and ML models' application.

2. Materials and Methods

3.1. Dataset

Fake News Dataset used in this study was accessed from the Information Security and Object Technology Research Lab (ISOT) at the University of Victoria in Canada [24]. The dataset consisted of two csv files corresponding to real and fake news collected between 2015 and 2018. The real news had 21417 rows and the fake news had 23481 rows. Both real and fake news had four features as columns being: the title, category, date, and actual news data all of which were string data. The close number of rows between the real and fake news showed that the dataset was balanced, and this saved the requirement for data imbalance treatment.

Dataset was explored in three stages being data pre-processing, dimension reduction and application of machine learning models. Fig. 1 shows a breakdown of the methodology adopted in the study.

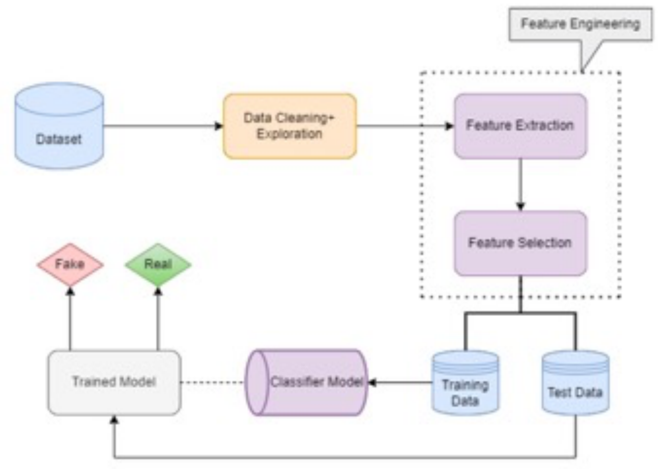


Fig. 1. Breakdown of the methodology adopted in this study.

3.2. Data pre-processing

Though the dataset was balanced, pre-processing of the data was essential to gain understanding on the characteristics of the dataset. In this respect data was checked for missing data and duplicate rows as well as class in balance. Then basic natural language processing (NLP) was applied using the NLTK toolkit, and included tokenisation, stop-word removal, stemming, lemmatization, feature extraction and feature selection [25, 26].

- For tokenisation, the NLTK provided work_tokenise() that split strings into individual words, and sent_tokenise() that split string into individual sentences.
- This was followed by removing stop words (e.g. 'the', 'is' and 'of') as these words interfere with the machine learning models accuracy by introducing noise.
- In addition, stemming converted words into their base form so to reduce the number of unique words in a text and improve model's efficiency and classification speed.
- Lemmatisation was similar to stemming in converting words to their bases, but it considered additionally the context of the word to generate more valid base form.
- Feature extraction was employed using Term Frequency-Inverse Document Frequency (TF-IDF), N-gram and counter vectorisation. TF-IDF generated weights for each feature in order to select features with high weights [27]. N-gram appointed adjacent sequence of n-words in the text [25, 26]. Counter vectorisation transformed textual into numerical data by creating a matrix that has each unique word represented by a column and each text sample by a row [25, 26].
- Feature selection enabled choosing the most important features in the model using Chi-squared test and univariate feature selection [28]. Both Chi-squared and univariate feature selection linked each feature to the output by calculating Chi-square value or p-value respectively [12].

3.3. Data analysis

Exploratory data analysis (EDA) was made prior to applying the ML models in order to understand patterns and relationships in the data. EDA allowed to gain understanding on the dataset used and detect any potential issues [29].

Considering the dataset main categories of real and fake news, binary classification using supervised ML models was applied. Real and fake news were classified as positive (class 1) or negative (class 0) [30]. In this respect, several ML models were used being logistic regression (LR), decision trees (DT), random forest (RF) and extreme gradient boost (XGBoost). LR was a commonly used linear model that separated between positive and negative classes based on probability ([10]. On the contrary, DT, RF and XGBoost separated positive from negative class based on non-linear modelling [9]. Models' evaluation was conducted by applying the common metrics related to accuracy, precision, recall, AUC-ROC and F1-score. These metrics were calculated based on true positives, true negatives, false positives and false negatives [22].

3. Results

3.1. Dataset Exploratory Data Analysis

Considering the close number of fake and real news in the dataset it was assumed to be balanced. Then features in dataset were reviewed to identify the most frequent categories. In this respect, the most frequent categories were political news and world news. Less common categories were 'news (unspecified)', 'politics', 'government news', 'left news', 'US-news' and 'Middle East'. Time series analysis of real and fake news showed that there were no incidents of true news between 2015 and 2016; where true news' prevalence increased post the end of 2017. Moreover, subject and date were excluded from the training dataset as they introduced bias in the classification [31].

Post-feature selection, univariate analysis was applied where the text and title columns were combined to form a single column. Univariate analysis showed that fake news were more prevalent on Twitter Hashtags while real news were more accessed via forums combined with Twitter. This latter finding suggested bias in the dataset. Subsequently, the '@' mentions were removed in order to reduce bias and make the dataset more suited for identifying fake news. In terms of the most frequent words in real and fake news, the results shows that most frequent words were fairly similar with 'trump' and 'said' being the top frequent words (Fig. 2). Likewise, word cloud analysis showed that the majority of true news were political, and the most encountered true news revolved around Trump, America and Reuters. This was the same as well for fake news that revolved around Trump and America.

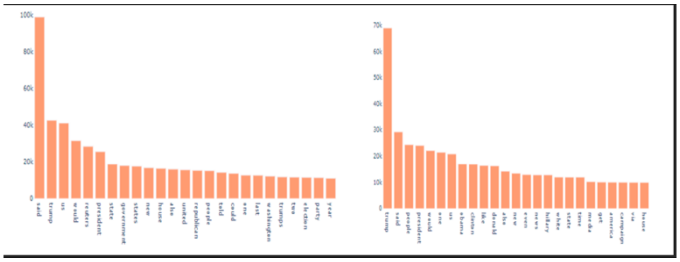
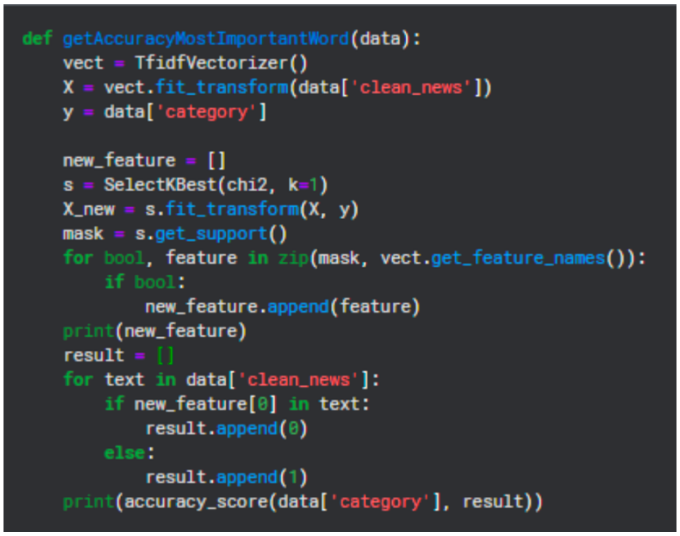


Fig. 2. Most frequent words in real (left) and fake (right) news.

Chi-squared analysis was then applied to identify the most relevant words because word frequency may not sufficiently discriminate real from fake news [12]. Chi-squared analysis indicated that 'said' and 'Reuters' were the most prominent words. Subsequently, a mock ML model was created for the word 'Reuters' to assess the impact of single prominent word on accuracy of classification. The model achieved accuracy of 99% that suggested that the model overfits the data relying heavily on the presence of 'Reuters' (Fig. 3).



```
def getAccuracyMostImportantWord(data):
    vect = TfidfVectorizer()
    X = vect.fit_transform(data['clean_news'])
    y = data['category']

    new_feature = []
    s = SelectKBest(chi2, k=1)
    X_new = s.fit_transform(X, y)
    mask = s.get_support()
    for bool, feature in zip(mask, vect.get_feature_names()):
        if bool:
            new_feature.append(feature)
    print(new_feature)
    result = []
    for text in data['clean_news']:
        if new_feature[0] in text:
            result.append(0)
        else:
            result.append(1)
    print(accuracy_score(data['category'], result))
```

Fig. 3. Model accuracy check for 'Reuters' word.

N-gram (bi-gram and tri-gram) analysis of true news, fake news and combined datasets showed the pattern in the language and structure of words (Fig. 4). This in turn improved the accuracy of ML models for fake news detection.

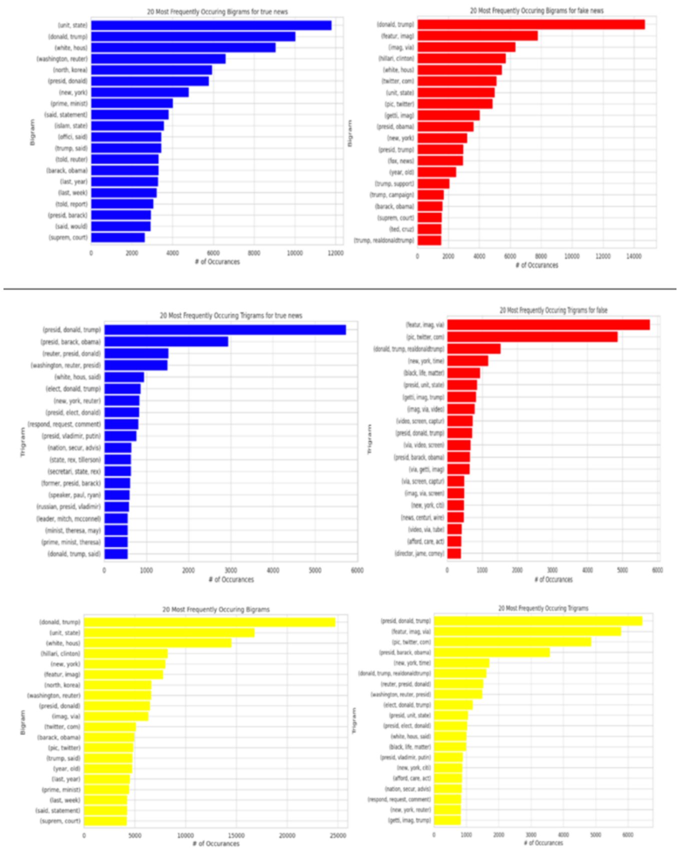


Fig. 4. Bi-gram plot (top) of real and fake news, tri-gram plot (middle) of real and fake news and tri-gram/bi-gram plots for combined (real and fake) news.

3.2. Machine Learning Models Evaluation

Evaluation of the ML models was applied in order to assess the models' performance and their generalisability to real world data. Evaluation metrics for assessing performance of the ML models included accuracy, precision, recall and F1-score. Evaluation was applied with three types of data. The first type included data that had not been subjected to feature selection; whereas, the second and third types included data that was subject to Chi-squared feature selection and univariate feature selection respectively (Table 1). It is worthnoting that all models showed high performance with metrics' values in the range of 92-99%. Of the four types of models, XGBoost showed the best overall performance, followed by RF and DT. LR on the other hand showed the least performance [22]. Moreover, when these models were used with bi-gram for feature extraction the accuracy obtained was high. Thus, chi-squared feature selection and bi-gram feature extraction produced the best accuracy when combined with ML models. Though RF had highest accuracy, XGBoost performance was con-sistent across all feature selection and extraction methods and that makes it the best model in analysing this particular dataset.

Table 1. Performance metrics of the machine learning models with different feature selection and extraction techniques.

Method	Model	Accuracy	Precision	Recall	F1-score
Performance metrics without feature selection					
TF-IDF with unigram	LR	0.9817	0.9755	0.9909	0.9832
	DT	0.9950	0.9951	0.9956	0.9953
	RF	0.9834	0.9776	0.9919	0.9847
	XGBoost	0.9965	0.9953	0.9982	0.9968
TF-IDF with bigram	LR	0.9805	0.9743	0.99	0.9821
	DT	0.9967	0.9961	0.9978	0.9969
	RF	0.9713	0.9615	0.9863	0.9738
	XGBoost	0.9968	0.9961	0.998	0.997
TF-IDF with trigram	LR	0.9774	0.9692	0.9895	0.9792
	DT	0.9971	0.9968	0.9978	0.9973
	RF	0.9719	0.962	0.9868	0.9742
	XGBoost	0.9967	0.9956	0.9982	0.9969
Counter vectorisation	LR	0.9923	0.9907	0.9951	0.9929
	DT	0.9943	0.9941	0.9953	0.9947
	RF	0.9855	0.9789	0.9946	0.9867
	XGBoost	0.9961	0.9949	0.998	0.9964
Counter vectorisation with filtered English words	LR	0.9936	0.97	0.9788	0.9744
	DT	0.9925	0.9283	0.9299	0.9291
	RF	0.9863	0.9574	0.9795	0.9683
	XGBoost	0.9942	0.9737	0.9834	0.9785
Performance metrics using Chi-squared feature selection					
TF-IDF with unigram	LR	0.9793	0.9729	0.9892	0.981
	DT	0.9934	0.9944	0.9934	0.9939
	RF	0.9954	0.9941	0.9973	0.9957
	XGBoost	0.996	0.9946	0.998	0.9963
TF-IDF with bigram	LR	0.9767	0.9699	0.9876	0.9786
	DT	0.995	0.9951	0.9956	0.9953
	RF	0.9965	0.9954	0.9983	0.9968
	XGBoost	0.9964	0.9956	0.9978	0.9967
TF-IDF with trigram	LR	0.9732	0.9641	0.9871	0.9754
	DT	0.9971	0.9968	0.9978	0.9973
	RF	0.9971	0.9961	0.9985	0.9973
	XGBoost	0.996	0.9947	0.998	0.9963
Counter vectorisation	LR	0.9919	0.9903	0.9949	0.9926
	DT	0.993	0.9937	0.9934	0.9935
	RF	0.9958	0.9941	0.998	0.9961
	XGBoost	0.996	0.9949	0.9978	0.9963
Counter vectorisation with filtered English words	LR	0.9925	0.9733	0.9776	0.9754
	DT	0.9926	0.9228	0.9269	0.9199
	RF	0.995	0.9613	0.981	0.971
	XGBoost	0.995	0.9756	0.9854	0.9805
Performance metrics using univariate feature selection					
	XGBoost				
TF-IDF with unigram	LR	0.9805	0.9739	0.9905	0.9821
	DT	0.9938	0.9939	0.9946	0.9943
	RF	0.9954	0.9948	0.9966	0.9957
	XGBoost	0.996	0.9944	0.9983	0.9963
TF-IDF with bigram	LR	0.9814	0.9746	0.9912	0.9828
	DT	0.9965	0.9959	0.9978	0.9968
	RF	0.9833	0.977	0.9924	0.9846
	XGBoost	0.9965	0.9958	0.9978	0.9968
TF-IDF with trigram	LR	0.9787	0.9715	0.9895	0.9804
	DT	0.9961	0.9949	0.9981	0.9964
	RF	0.9792	0.972	0.99	0.9809
	XGBoost	0.9962	0.9946	0.9982	0.9964
Counter vectoriser	LR	0.9917	0.9905	0.9941	0.9923
	DT	0.9925	0.9936	0.9924	0.993
	RF	0.995	0.9925	0.9983	0.9954
	XGBoost	0.996	0.9949	0.9978	0.9963
	LR	0.993	0.9676	0.9764	0.972
	DT	0.9925	0.9254	0.9301	0.9278
	RF	0.9947	0.9651	0.9781	0.9716
	XGBoost	0.995	0.9737	0.9827	0.9782

It is worthnoting to mention that feature selection reduced the number of features allowing the inclusion of only the most important and relevant ones to the dataset (fake news in this case) [22, 23]. This is very important in order to avoid overfitting and decreasing the running time of the model [22, 23]. Likewise, feature classification points out the most important features in a model. The best metrics were observed for bi-gram and chi-squared when applied to RF model as mentioned above. This was also apparent in the confusion matrix and AUC-ROC plot (Fig. 5).

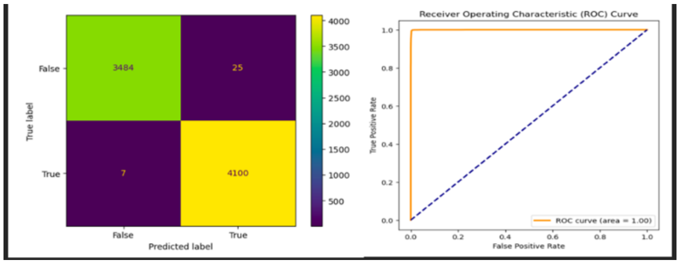


Fig. 5. Confusion matrix (left) and AUC-ROC (right) of RF model applied after chi-squared feature selection and bi-gram feature extraction.

4. Conclusions

The present study evaluated machine learning algorithms with different feature selection and extraction techniques for detecting fake news. Out of the evaluated machine learning algorithms, RF showed to be the best performing model when combined with chi-squared feature selection and bi-gram extraction. Hence, the accuracy, precision, recall and F1-score for this model were all above 99.5%; and that showed that it outperformed all the other models. Yet it is worth mentioning that XGBoost model sowed high accuracy and robustness, so it was the second most performing model after RF. Moreover, this study highlighted the important of feature extraction in text classification as it reduces the number of variables and avoids overfitting of the mod-el. Moreover, the feature extraction methods captured the most important features in the method for text classification. Therefore, the study emphasised the importance of feature selection and feature extraction in text classification models.

References

[1] Appel, M., Marker, C., & Gnambs, T. (2020). Are social media ruining our lives? A re-view of meta-analytic evidence. Review of General Psychology, 24(1), 60-74.

[2] Geng, R., Wang, S., Chen, X., Song, D., & Yu, J. (2020). Content marketing in e-commerce platforms in the internet celebrity economy. Industrial Management & Data Systems, 120(3), 464-485.

[3] Xu, K., Wang, F., Wang, H., & Yang, B. (2019). Detecting fake news over online social media via domain reputations and content understanding. Tsinghua Science and Tech-nology, 25(1), 20-27.

[4] Guo, Z. (2023). Understanding and Combating Online Social Deception (Doctoral dissertation, Virginia Tech).

[5] Collins, B., Hoang, D. T., Nguyen, N. T., & Hwang, D. (2021). Trends in combating fake news on social media–a survey. Journal of Information and Telecommunication, 5(2), 247-266.

[6] Guo, C., Cao, J., Zhang, X., Shu, K., & Yu, M. (2019). Exploiting emotions for fake news detection on social media. arXiv preprint arXiv:1903.01728.

[7] Shahid, W., Li, Y., Staples, D., Amin, G., Hakak, S., & Ghorbani, A. (2022). Are you a cyborg, bot or human?—a survey on detecting fake news spreaders. IEEE Access, 10, 27069-27083.

[8] Coombes, R., & Davies, M. (2022). Facebook versus The BMJ: when fact checking goes wrong. bmj, 376.

[9] Hakak, S., Alazab, M., Khan, S., Gadekallu, T. R., Maddikunta, P. K. R., & Khan, W. Z. (2021). An ensemble machine learning approach through effective feature extraction to classify fake news. Future Generation Computer Systems, 117, 47-58.

[10] Hansrajh, A., Adeliyi, T.T. and Wing, J., 2021. Detection of online fake news using blending ensemble learning. Scientific Programming, 2021, pp.1-10.

[11] Vijayaraghavan, S., Wang, Y., Guo, Z., Voong, J., Xu, W., Nasseri, A., ... & Wadhwa, E. (2020). Fake news detection with different models. arXiv preprint arXiv:2003.04978.

[12] Fayaz, M., Khan, A., Bilal, M., & Khan, S. U. (2022). Machine learning for fake news classification with optimal feature selection. Soft Computing, 26(16), 7763-7771.

[13] Elsaeed, E., Ouda, O., Elmogy, M. M., Atwan, A., & El-Daydamony, E. (2021). Detect-ing fake news in social media using voting classifier. IEEE Access, 9, 161909-161925.

[14] Garcia, G. L., Afonso, L. C., Passos, L. A., Jodas, D. S., da Costa, K. A., & Papa, J. P. (2023). FakeRecogna Anomaly: Fake News Detection in a New Brazilian Corpus. In VISIGRAPP (4: VISAPP) (pp. 830-837).

[15] Aldwairi, M., & Alwahedi, A. (2018). Detecting fake news in social media networks. Procedia Computer Science, 141, 215-222.

[16] Yang, Y., Zheng, L., Zhang, J., Cui, Q., Li, Z., & Yu, P. S. (2018). TI-CNN: Convolu-tional neural networks for fake news detection. arXiv preprint arXiv:1806.00749.

[17] Bahad, P., Saxena, P., & Kamal, R. (2019). Fake news detection using bi-directional LSTM-recurrent neural network. Procedia Computer Science, 165, 74-82.

[18] Sarnovský, M., Maslej-Krešňáková, V., & Ivancová, K. (2022). Fake news detection related to the covid-19 in slovak language using deep learning methods. Acta Poly-technica Hungarica, 19(2), 43-57.

[19] Chen, M. Y., Lai, Y. W., & Lian, J. W. (2023). Using deep learning models to detect fake news about COVID-19. ACM Transactions on Internet Technology, 23(2), 1-23.

[20] Alhakami, H., Alhakami, W., Baz, A., Faizan, M., Khan, M. W., & Agrawal, A. (2022). Evaluating Intelligent Methods for Detecting COVID-19 Fake News on Social Media Platforms. Electronics, 11(15), 2417.

[21] Amer, E., Kwak, K. S., & El-Sappagh, S. (2022). Context-based fake news detection model relying on deep learning models. Electronics, 11(8), 1255.

[22] Capuano, N., Fenza, G., Loia, V., & Nota, F. D. (2023). Content Based Fake News De-tection with machine and deep learning: a systematic review. Neurocomputing.

[23] Mishra, S., Shukla, P., & Agarwal, R. (2022). Analyzing machine learning enabled fake news detection techniques for diversified datasets. Wireless Communications and Mobile Computing, 2022, 1-18.

[24] Information Security and Object Technology Research Lab, 2022. Available at: https://onlineacademiccommunity.uvic.ca/isot/ Accessed: 28/08/2023.

[25] Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A Python natural language processing toolkit for many human languages. arXiv preprint arXiv:2003.07082.

[26] Hardeniya, N., Perkins, J., Chopra, D., Joshi, N., & Mathur, I. (2016). Natural language processing: python and NLTK. Packt Publishing Ltd.

[27] Christian, H., Agus, M. P., & Suhartono, D. (2016). Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF). ComTech: Computer, Mathematics and Engineering Applications, 7(4), 285-294.

[28] Laborda, J., & Ryoo, S. (2021). Feature selection in a credit scoring model. Mathemat-ics, 9(7), 746.

[29] Meyn, L. F. (2018). Fake news prediction on facebook: Design and implementation of a fake news prediction tool (Doctoral dissertation, University of Nebraska at Omaha).

[30] Sharma, M. K., Kumar, P., Rasool, A., Dubey, A., & Mahto, V. K. (2021, November). Classification of actual and fake news in pandemic. In 2021 Fifth International Con-ference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) (pp. 1168-1174). IEEE.

31. Bozarth, L., & Budak, C. (2020, May). Toward a better performance evaluation frame-work for fake news classification. In Proceedings of the international AAAI confer-ence on web and social media (Vol. 14, pp. 60-71).