Disaster Analysis Through Tweets

Chapter · March 2023

DOI: 10.1007/978-981-19-9225-4_40

6 authors, including:

Anshul Sharma
Chandigarh University
29 PUBLICATIONS **95** CITATIONS
SEE PROFILE

Divneet Singh Kapoor
Chandigarh University
48 PUBLICATIONS **436** CITATIONS
SEE PROFILE

Kiran Jot Singh
33 PUBLICATIONS **411** CITATIONS
SEE PROFILE

Disaster Analysis Through Tweets

Anshul Sharma, Khushal Thakur, Divneet Singh Kapoor, Kiran Jot Singh, Tarun Saroch, and Raj Kumar

Abstract Social media has assumed a huge part in scattering data about these disasters by permitting individuals to share data and request help. During disaster, social media gives a plenty of data which incorporates data about the idea of disaster, impacted individuals' feelings and aid ventures. This data proliferated over the social media can save great many life by alarming others, so they can make a hesitant move. Numerous offices are attempting to automatically dissect tweets and perceive disasters and crises. This sort of work can be advantageous to a great many individuals associated with the Web, who can be alarmed on account of a crises or disaster. Twitter information is unstructured information; in this manner, natural language processing (NLP) must be performed on the Twitter information to arrange them into classes–"Connected with Disaster" and "Not connected with Disaster." The paper does an expectation on the test set made from the first informational collection. It does an exactness testing of the classifier model created. This paper involves Naive Bayes classification mechanism for building the classifier model and for making predictions.

Keywords Twitter analysis · Natural language processing · Classifier model · Naive Bayes classification

1 Introduction

Nearly, 2.9 trillion people were affected by natural disasters during the period 2000 and 2012 causing damages exceeding $1.7 trillion. Damages in 2011 set the record high, reaching whoopingly $371 billion while in 2012 for the third consecutive year the damages crossed $100 billion [1]. Research through Internet-based media in

A. Sharma · K. Thakur (✉) · D. S. Kapoor · K. J. Singh
Electronics and Communication Engineering Department, Chandigarh University, Mohali, Punjab 140413, India
e-mail: khushal.ece@gmail.com

T. Saroch · R. Kumar
Computer Science and Engineering Department, Chandigarh University, Mohali, Punjab 140413, India

S. Kumar et al. (eds.), *Third Congress on Intelligent Systems*, Lecture Notes in Networks and Systems 608, https://doi.org/10.1007/978-981-19-9225-4_40

disaster occasions is growing, more expressively, interest in miniature publishing the content to a blog, in case of emergencies are on the climb. Early investigation shows that essential modern and on the spot updates can be found in miniature blog messages about a spreading out crisis, hurrying a premium in overwhelming handling and insight mechanical assemblies. Social media are intuitive innovations that work with the creation and sharing of data, thoughts, interests, and different types of articulation through virtual communities and networking [2]. These days there are numerous social media stages like Facebook, Instagram, Twitter, YouTube, Blogger, and so forth and this undertaking just work on Twitter. Thus, Twitter is a well-known miniature publishing content to a blog social media benefits that permits client to share short messages called tweets which are basically of 160 words or less [3].

In the era of social media, miniature online journals are frequently used to share data and track overall population assessment during any human-influencing occasion, for example, sports matches, political decisions, natural disasters, and so on; many relief organizations screen this sort of information routinely to distinguish disasters. This sort of work might be useful to a huge number of clients on the Web, who can be alarmed on account of a crises or disaster. In any case, it is outside the realm of possibilities for people to physically take a look at the mass measure of information and recognize disasters in real time. For this reason, many examination works have been proposed to introduce words in machine-understandable portrayals and apply AI strategies on the word portrayals to distinguish the opinion of a text.

Work by Saroj and Pal [4] principally implies two natural perils grassfires in Oklahoma and red stream flood. Information for examination were acquired through Twitter search API. The catchphrases #redriver and #redriver were utilized to get red stream flood tweets though #Oklahoma, #okfire, #grassfire were utilized to recover Oklahoma grassfire tweets. Goswami and Raychaudhuri [5] had accomplished a similar work for identification of disaster tweets utilizing natural language processing, and their end-product incorporates the accuracy of 71.5%. Vieweg had done his research for mainly two disasters, i.e., grassfires and red stream flood, whereas Shriya and Debaditya have done the same research, but were able to achieve the accuracy of 71.5% or 0.715. Furthermore, advances in technologies like Internet of Things, wireless sensor networks, and computer vision can be used to develop newer multi-domain solutions [6–11].

Unlike Vieweg research, the work presented in the paper is designed for all types of disaster, and this work additionally has a principle objective of expanding the accuracy with thinking about the other performance metrics. Present paper presents natural language processing with disaster tweets where the main objective has been kept to identify whether a specific tweet is about a real disaster or not.

Present paper is coordinated as follows: Section 2 introduces the foundation of dataset. Section 3 introduces proposed system. Section 4 introduces the outcome of the present work. Section 5 concludes and introduce future extent of our concern proclamation.

Table 1 Training dataset headers

	Id	Keyword	Location	Text	Target
0	0	NaN	NaN	Our deeds are the reason of this #earthquake May ALLAH forgive us all	1
1	2	NaN	NaN	Forest fire near La Ronge Sask. Canada	1
2	3	NaN	NaN	All residents asked to "shelter in place" are being notified by officers. No other evacuation or shelter in place orders are expected	1
3	9	NaN	NaN	13,000 people receive #wildfires evacuation orders in California	1
4	11	NaN	NaN	Just got sent this photo from Ruby #Alaska as smoke from #wildfires pours into a school	1

2 Background of Datasets

Normal language processing (NLP) is a part of artificial intelligence that helps PCs comprehend, decipher, and use human dialects. NLP permits PCs to speak with individuals, utilizing a human language. Normal language processing likewise furnishes PCs with the capacity to understand text, hear discourse, and decipher it. For this venture, NLP for feeling examination or text understanding is utilized [12]. The datasets utilized for this task are taken from Kaggle [13] with more than 10,000 tweets extracted from Twitter through Twitter API.

Train informational index contains four elements, i.e., ID, area, text, and target, as shown in Table 1. Location is the component where area for the tweet is given. Text highlight contains the genuine text of the tweet got from Twitter. The target is the element of a dataset concerning which you need to acquire a more profound arrangement. An administered AI calculation utilizes verifiable information to learn designs [14] and uncover connections between different elements of your informational index and the objective.

For test dataset, there are three elements, i.e., ID, location, and text, as shown in Table 2. ID highlights contain the essential ID utilized for recognizable proof of the tweet; anyway for this work, there is no prerequisite for this part, and in future, it will be dropped. Location contains the area of the real tweet. Text includes the tweet only, same as in train dataset.

3 Proposed Framework

In this section, we discuss in detail the various processes of our system methodology. The system is composed of three different modules:

1. Extraction of data,
2. Sorting of data, and
3. Analysis of data.

Table 2 Test dataset headers

	Id	Keyword	Location	Text
0	0	NaN	NaN	Just happened a terrible car crash
1	2	NaN	NaN	Heard about #earthquake is different cities, stay safe everyone
2	3	NaN	NaN	There is a forest fire at spot pond; geese are fleeing across the street, I cannot save them all
3	9	NaN	NaN	Apocalypse lighting. #Spokane #wildfires
4	11	NaN	NaN	Typhoon Soudelor kills 28 in China and Taiwan

a. **Data extraction**

Data extraction is mainly done with the help of Twitter API that will provide data for the project in CSV format. Then, it will be easier to fetch and modify data according to the problem statement.

b. **Sorting of data**

There are so many superfluous/undesirable text or images, accordingly this interaction requires pre-handling. Need to eliminate #hashtags, numbers, halting words, (for example, a, the, an, there, and so forth), images, and so on; there should be likelihood for similar tweets due to retweets or copies in information. This can be taken out by utilizing specific orders accessible in pandas. The put away tweets are currently examined, and their calamity not really settled [15]. This is finished by really looking at each tweet against a bunch of predefined gauged keywords. The tweets which don't have a place with any class are promptly disposed of. The tweets which don't have a place with any classification are promptly disposed of.

c. **Analysis of data**

Investigation of information is most significant angle for this issue articulation. This table contains some extraordinary word reference of words which will look for the words in the tweets. Based on looking and matching, tweets will be considered as a genuine or fake disaster tweets. Model tuning permits to alter models, so they create the most dependable results and give profoundly important experiences into information, empowering us to settle on the best choices. Then, at that point, foreseeing out the best models out of them with most noteworthy precision by contrasting various models like Naive Bayes, extra tree classifier, random forest, and so forth and evaluating distinctive execution boundaries like normalization, feature selection, and some outlier removal.

3.1 Pseudo-code

Step 1: Start
Step 2: Installing PyCaret [16] on the machine.
Step 3: Install/import other necessary packages (Mentioned below):

- NLTK—All, stopwords
- Pandas, Numpy, OS
- Warnings [17]—simplefilter/ignore
- MatPlotlib [18]—matplotlib/pyplot

Step 4: Import Train Dataset using Pandas.
Step 5: Clearing Tweet text (tweets from train dataset) and dropping unnecessary columns.
Step 5.1: Removing punctuation
Step 5.2: Removing StopWords
Step 6: Building Machine Learning model.
Step 6.1: importing classification module.
Step 6.2: Setting up the environment.
Step 6.3: Building different classifier models (Naive Bayes, Extra tree model).
Step 6.4: Compare models.
Step 7: Evaluate Model.
Step 7.1: Building Confusion matrix for Naive Bayes classifier.
Step 7.2: Building Confusion matrix for Extra Tree classifier.
Step 7.3: Calculating different performances matrices.
Step 8: Import the Test dataset using Pandas.
Step 9: Again clearing tweet text and dropping unnecessary columns.
Step 9.1: Removing punctuation
Step 9.2: Removing StopWords
Step 10: Creating a prediction model.
Step 11: Saving the model.
Step 12: Load the model.
Step 13: Analyze the model by plotting different charts

4 Algorithm Used for Classification

4.1 Naive Bayes Classifier

Naive Bayes classifiers are a grouping of calculation or estimations subject to Bayes' theorem. They are among the most straightforward Bayesian association models, yet combined with piece thickness assessment, they can accomplish higher exactness levels. It is not a single algorithm but a group of calculations where every one of them share a typical standard, for example, each pair of elements being characterized

is free of one another. They are among the easiest Bayesian organization models, yet combined with piece thickness assessment, they can accomplish higher exactness levels. Beside this classifier, another classifier known as extra-tree classifier is also used. As both are having same accuracy but considering other factors like time taken, recall was not comparable to Naive Bayes classifier that is why extra-tree classifier is neglected.

4.2 Results

Using different classifiers model from PyCaret package, the best model suited for this project was found to be the Naive Bayes classifier with SD and mean accuracy of 0.0248 and 0.7347, respectively, as shown in Table 3. This was the most noteworthy among the wide range of various classifiers, for example, extra-tree classifier, *K*-nearest neighbor classifier have the mean accuracy of 0.7347 and 0.7118, respectively, and others as shown below. Using different classifiers model from PyCaret package, the best model suited for this project was found to be the Naive Bayes classifier with SD and mean accuracy of 0.0248 and 0.7347, respectively. This was the most noteworthy among the wide range of various classifiers, for example, extra-tree classifier, *K*-nearest neighbor classifier have the mean accuracy of 0.7347 and 0.7118, respectively, and others as shown below.

Then, different plots like AUC-ROC curves, boundary plot, and precision-recall have been plotted for Naive Bayes classifier.

AUC-ROC curve depicts the rate of correctness of the classification model used in the project, i.e., Naive Bayes classification model. Figure 1 depicts that rate for both true and fake disaster tweets is 0.79.

As shown in Fig. 2, the initial point (0, 0), the threshold is set to 1.0 so that there would be no discrimination both true or fake disaster tweets. And, at final point (1, 1), the threshold is set to 0.0 so that there will be classification between true and fake disaster tweets. More the curve curved toward (1, 1), more is the precision, and we have to make the curve more toward (1, 1). For evaluating our model, confusion matrix is developed as shown in Fig. 3.

$$\text{From confusion matrix, Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} = \frac{620 + 1037}{620 + 1037 + 347 + 287} = 0.7232$$

$$\text{From confusion matrix, Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{620}{620 + 347} = 0.6411$$

$$\text{From confusion matrix, Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{620}{620 + 280} = 0.6888$$

Table 3 Best fit model evaluation

	Model	Accuracy	AUC	Recall	Prec	$F1$	Kappa	MCC	TT (s)
nb	Naive Bayes	0.7347	0.7937	0.694	0.6935	0.6933	0.4596	0.4601	0.473
et	Extra tree classifier	0.7347	0.7666	0.5625	0.7627	0.6464	0.442	0.4558	22.47
rf	Random forest classifier	0.7315	0.7777	0.5252	0.7836	0.6281	0.4306	0.4523	15.428
dt	Decision tree classifier	0.7118	0.6994	0.6081	0.6898	0.6452	0.4044	0.4075	2.164
knn	K neighbors classifier	0.6915	0.7379	0.6029	0.6567	0.628	0.3654	0.3669	4.585
gbc	Gradient boosting classifier	0.6731	0.7343	0.3295	0.7987	0.4641	0.284	0.3426	28.016
ada	AdaBoost classifier	0.6675	0.7059	0.296	0.8291	0.4328	0.2666	0.3387	7.223
qda	Quadratic discriminant classifier	0.6288	0.5748	0.1758	0.8509	0.287	0.1649	0.2592	41.663
lr	Logistic regression	0.6232	0.6182	0.2494	0.4491	0.2937	0.162	0.1755	2.509
dummy	Dummy classifier	0.5676	0.5	0	0	0	0	0	0.083
ridge	Ridge classifier	0.5646	0	0.069	0.2404	0.0976	0.012	0.0166	29.943
svm	SVM—linear kernel	0.4731	0	0.6866	0.302	0.0.4194	− 0.003	− 0.028	7.192
lda	Linear discriminant analysis	0.2249	0.232	0.2175	0.2001	0.2079	0.0479	0.0481	96.072

GuassianNB (priors = None, var_smoothing = 1e − 09)

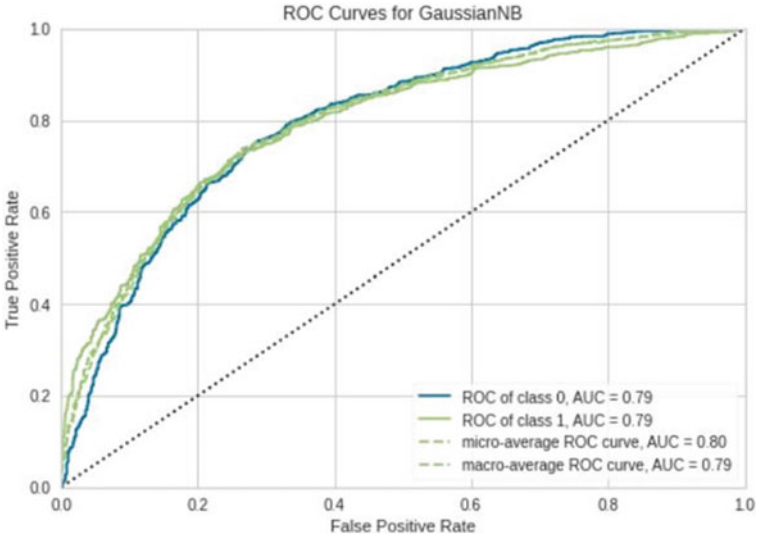


Fig. 1 ROC curves for Gaussian NB

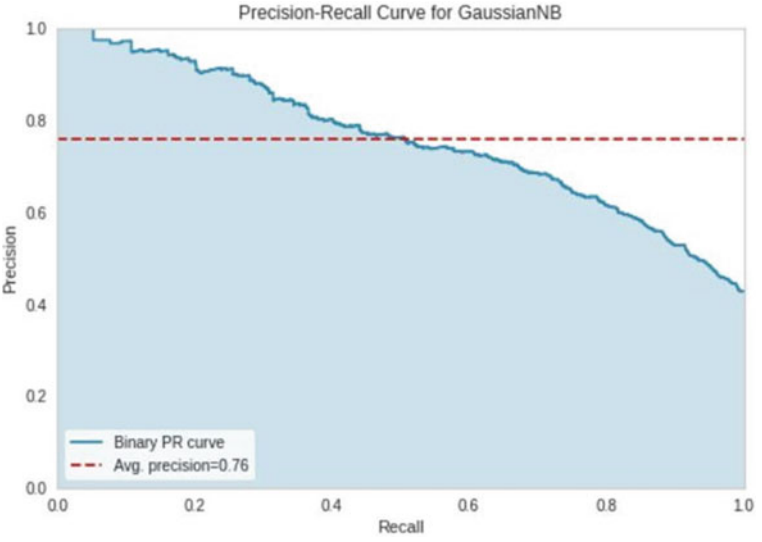
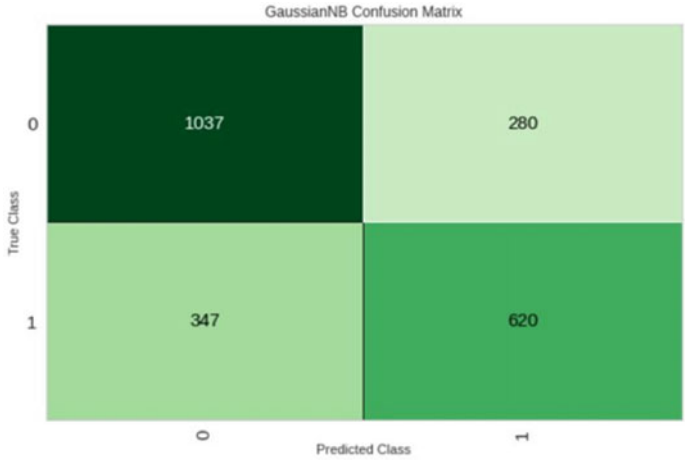


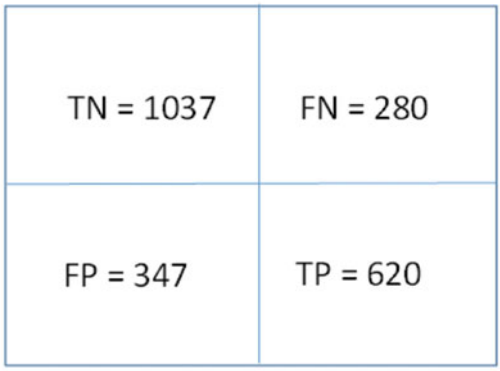
Fig. 2 Precision recall curve for Gaussian NB

$$\text{Now, } F1\text{Score} = \frac{2 \times (\text{Precision} \times \text{Recall})}{(\text{Precision} + \text{Recall})} = \frac{2 \times (0.6411 \times 0.6888)}{(0.6411 + 0.6888)} = 0.6640$$

Conclusion obtained from confusion matrix.



(a)



(b)

Fig. 3 **a** Confusion matrix for Gaussian NB **b** confusion matrix with values

As can be seen from Table 4, all of them are very close to the result obtained while building model or while comparing model. After creation and tuning of the training dataset, the trained model was used for making predictions. PyCaret builds a pipeline of all the steps and will pass the unseen data into the pipeline and give us the results. Figure 4 shows classification of dataset.

As per Fig. 5, among the many calamities, fire received the most tweets. Other catastrophes (whether man-made or natural) include disaster like bombing, accident, flames, etc. Although word clouds may have been used to express the same thing, this graph worked better for this assignment.

Table 4 Performance of Gaussian NB

Parameter	In value	In percentage (%)
Accuracy	0.7232	72.32
Precision	0.6411	64.11
Recall	0.6888	68.88
$F1$-score	0.6640	66.40

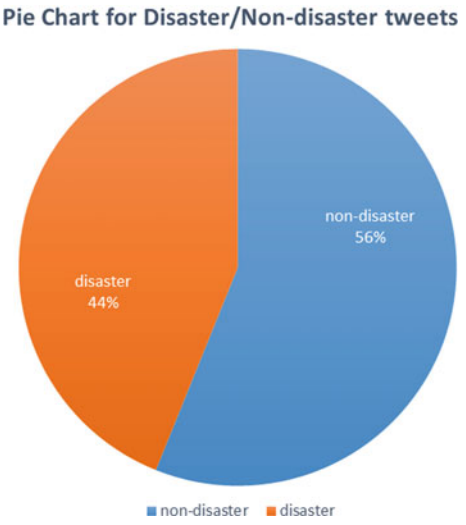


Fig. 4 Classification of tweets into disaster/non-disaster

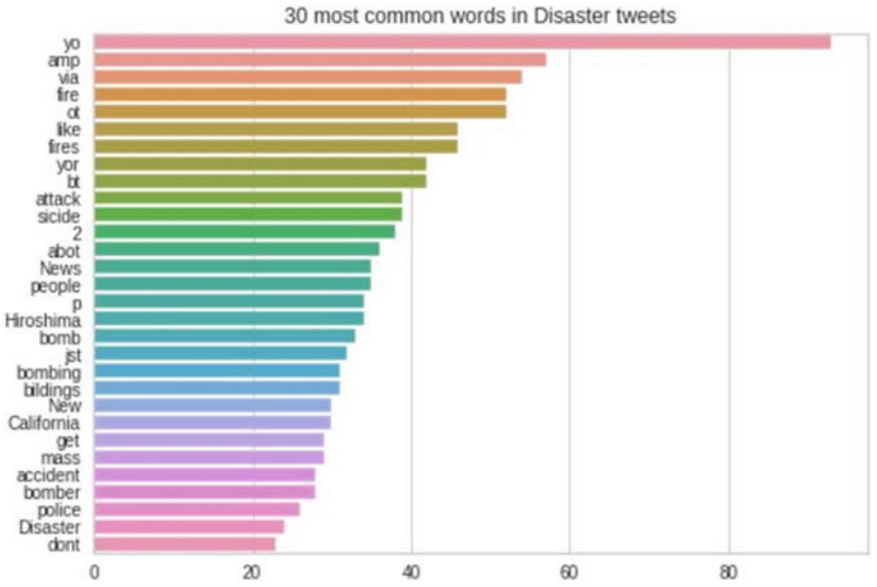


Fig. 5 Most commonly used words in disaster tweets

5 Conclusion

This task depends on natural language processing (NLP). This task sorts a specific tweet is a real disaster or not. This can be valuable to countless clients on the Web, who can be alarmed on account of a crises or disaster. This work successfully achieved the increment in accuracy to 0.7373. This task as finally working with 0.7303 of precision. It had anticipated 10,875 tweets which has 56.1% non-disaster tweets, and rest 43.9% contains disaster tweets (analyzed in this task). The most common form of disaster found with this analysis is fire.

The primary target of this task was to build the accuracy regarding the other performance metrics. There might be an amazing chance to build the precision while working on different boundaries too. With expansion in an Earth-wide temperature boost, there might be the chance of the disaster in a particular region, so this task can help us with seeing whether or not a tweet is a genuine calamity tweet. This task doesn't manages any sort of area or geo-planning. In future, this can be created utilizing diverse AI and machine learning bundles.

References

1. Rayer Q et al. (2021) Water ınsecurity and climate risk: ınvestment ımpact of floods and droughts. Palgrave Stud Sustain Bus Assoc with Futur Earth 157–188. https://doi.org/10.1007/978-3-030-77650-3_6
2. Miño-Puigcercós R et al. (2019) Virtual communities as safe spaces created by young feminists: ıdentity, mobility and sense of belonging. Identities, Youth Belong 123–140. https://doi.org/10.1007/978-3-319-96113-2_8
3. Greco F, Polli A (2020) Emotional text mining: customer profiling in brand management. Int J Inf Manage 51:101934. https://doi.org/10.1016/J.IJINFOMGT.2019.04.007
4. Saroj A, Pal S (2020) Use of social media in crisis management: a survey. Int J Disaster Risk Reduct 48:101584. https://doi.org/10.1016/J.IJDRR.2020.101584
5. Goswami S, Raychaudhuri D (2020) Identification of disaster-related tweets using natural language processing. In: International conference on recent trends in artificial ıntelligence, Iot, smart cities & applications (ICAISC-2020), SSRN Electron J https://doi.org/10.2139/SSRN.3610676
6. Jawhar Q et al (2020) Recent advances in handling big data for wireless sensor networks. IEEE Poten 39(6):22–27. https://doi.org/10.1109/MPOT.2019.2959086
7. Jawhar Q, Thakur K (2020) An improved algorithm for data gathering in large-scale wireless sensor networks. Lect Notes Electr Eng 605:141–151. https://doi.org/10.1007/978-3-030-30577-2_12
8. Sachdeva P, Singh KJ (2016) Automatic segmentation and area calculation of optic disc in ophthalmic images. In: 2015 2nd International Conference Recent Advance Engineering Computer Science RAECS 2015. https://doi.org/10.1109/RAECS.2015.7453356
9. Sharma A et al. (2022) Exploration of IoT nodes communication using LoRaWAN in forest environment. Comput Mater Contin 71(2):6240–6256. https://doi.org/10.32604/CMC.2022.024639
10. Sharma A, Agrawal S (2012) Performance of error filters on shares in halftone visual cryptography via error diffusion. Int J Comput Appl 45:23–30

11. Singh K et al. (2014) Image retrieval for medical imaging using combined feature fuzzy approach. In: 2014 International conference devices, circuits communication ICDCCom 2014—proceedings. https://doi.org/10.1109/ICDCCOM.2014.7024725
12. Chakravarthi BR et al. (2022) DravidianCodeMix: sentiment analysis and offensive language identification dataset for dravidian languages in code-mixed text. Lang Resour Eval 1–42. https://doi.org/10.1007/S10579-022-09583-7/TABLES/15
13. QaziUmair et al (2020) GeoCoV19. SIGSPATIAL Spec 12(1):6–15. https://doi.org/10.1145/3404820.3404823
14. Priyanka EB et al (2022) Digital twin for oil pipeline risk estimation using prognostic and machine learning techniques. J Ind Inf Integr 26:100272. https://doi.org/10.1016/J.JII.2021.100272
15. Vera-Burgos CM, Griffin Padgett DR (2020) Using twitter for crisis communications in a natural disaster: hurricane harvey. Heliyon 6(9):e04804. https://doi.org/10.1016/J.HELIYON.2020.E04804
16. GitHub—pycaret/pycaret: an open-source, low-code machine learning library in Python. https://github.com/pycaret/pycaret. Last accessed 05 April 2022
17. Warnings—Warning control—Python 3.10.4 documentation. https://docs.python.org/3/library/warnings.html. Last accessed 05 April 2022
18. Matplotlib documentation—Matplotlib 3.5.2 documentation. https://matplotlib.org/stable/. Last accessed 05 April 2022