

Prediction Of COVID-19 Severity by Applying Machine and Deep Learning Techniques

Narahariseti Keerthi

PG scholar, Department of MCA, CDNR collage, Bhimavaram, Andhra Pradesh.

A.Naga Raju

(Assistant Professor), Master of Computer Applications, DNR collage, Bhimavaram, Andhra Pradesh.

Abstract: This project focuses on predicting the severity of COVID-19 by applying machine learning and deep learning techniques to processed textual and numerical data. The dataset, enhanced with a "Used Technologies" label, includes country-wise case statistics and descriptive information related to COVID-19. Natural Language Processing (NLP) techniques using NLTK were applied to clean and analyze text data. The textual information was then transformed into numerical form using TF-IDF vectorization, enabling the integration of text-based insights with structured numerical data. Subsequent analysis involved computing the distribution of technologies mentioned, and aggregating critical, active, and new case counts for each country. The dataset was split into training and testing subsets, and a Random Forest algorithm was implemented to predict COVID-19 severity indicators based on the available features. The trained model achieved an accuracy of 80%, as verified through confusion matrix evaluation. The study demonstrates the potential of combining text mining, statistical analysis, and machine learning models to predict the severity and spread of COVID-19. These insights can aid in early detection, policy planning, and resource allocation during pandemic scenarios.

Introduction

The COVID-19 pandemic posed an unprecedented global health challenge, leading to millions of infections and significant social and economic disruption. As the virus rapidly spread across countries, early prediction of its severity in patients and across regions became essential for effective healthcare response and resource management. With the increasing availability of real-time data and advancements in artificial intelligence, machine learning (ML) and deep learning (DL) models have emerged as powerful tools to analyze

complex datasets and generate accurate predictions. These technologies offer the potential to forecast critical aspects such as the likelihood of severe cases, geographical spread, and healthcare resource needs.

In this project, we explore a data-driven approach that leverages natural language processing (NLP) techniques along with ML algorithms to predict the severity of COVID-19 cases. Textual data related to COVID-19 was preprocessed using NLTK and converted into TF-IDF vectors to extract meaningful patterns. We then performed statistical analyses on country-wise data including new, active, and critical cases. A Random Forest classifier was trained on the processed dataset to predict the technologies used and infer potential severity trends. The model achieved an accuracy of 80%, highlighting the effectiveness of ML in providing insights into pandemic trends and supporting informed decision-making.

Literature Survey

COVID-19 Severity Prediction Using Machine Learning Algorithms

In this study, researchers used patient health records, including symptoms, comorbidities, and demographic information, to train machine learning models for predicting the severity of COVID-19. Techniques such as Decision Trees, Support Vector Machines, and Random Forests were applied. Among them, Random Forest achieved the highest accuracy in classifying patients into mild, moderate, or severe categories. This work demonstrated the feasibility of using ML for assisting in early triage and resource allocation during the pandemic.

Deep Learning Models for COVID-19 Diagnosis Using Chest X-rays

A deep learning-based approach was proposed using Convolutional Neural Networks (CNNs) to analyze chest X-ray images and detect COVID-19 infection levels. The model leveraged transfer learning with pre-trained architectures like VGG16 and ResNet50. Results showed that CNNs can accurately classify severe COVID-19 cases with minimal preprocessing, highlighting the potential of deep learning in medical image analysis for rapid diagnosis and severity assessment.

Natural Language Processing for COVID-19 Research: A Review

This review paper explored how NLP techniques were applied to COVID-19-related textual data, including social media, scientific articles, and clinical notes. It discussed methods like sentiment analysis, topic modeling, and TF-IDF for extracting useful insights. The paper emphasized how NLP helps identify public sentiment, misinformation trends, and even symptom descriptions, which can feed into predictive models for better pandemic response.

Predicting Hospitalization Needs of COVID-19 Patients Using AI

A research team developed a machine learning system to predict whether COVID-19 patients would require hospitalization based on early symptom data and comorbidities. Using logistic regression and ensemble learning techniques, the system achieved over 85% accuracy in predicting hospitalization needs. The study underlined how predictive analytics can help manage hospital capacities during health crises.

Hybrid Machine Learning Models for COVID-19 Risk Prediction

This paper proposed a hybrid approach combining machine learning and statistical methods to predict COVID-19 risk at both individual and community levels. It used data such as testing rates, mobility

patterns, and demographic information. By integrating Random Forest with time-series forecasting models, it achieved robust predictions and was able to identify high-risk zones effectively. The hybrid model showed superior performance compared to single algorithm methods.

Existing Method

Existing methods for predicting the severity of COVID-19 primarily rely on traditional statistical techniques and standard machine learning algorithms. These approaches often use structured clinical data such as age, gender, symptoms, comorbidities, and vital signs to train models like Logistic Regression, Decision Trees, and Support Vector Machines. Additionally, some methods incorporate textual data using basic Natural Language Processing (NLP) techniques like Bag-of-Words or TF-IDF to extract relevant features. However, these methods often fall short when dealing with complex, high-dimensional data or when required to capture deep contextual meaning from text. Furthermore, most traditional systems lack integration of both structured numeric data and unstructured textual data, limiting their predictive performance in real-world scenarios where a combination of both is essential for accurate severity classification.

Proposed Method

The proposed method integrates Natural Language Processing (NLP) with machine learning techniques to enhance the prediction of COVID-19 severity by utilizing both textual and numerical data. Initially, text data related to COVID-19 is preprocessed using NLTK, including tokenization, stop word removal, and lemmatization. The cleaned text is then converted into numerical form using TF-IDF vectorization, allowing for meaningful representation of term importance. Alongside this, numerical features such as country-wise counts of new, active, and critical cases are analyzed. The combined feature set is used to train a Random Forest classifier, which learns patterns and correlations to accurately predict the severity and

associated technologies used in handling COVID-19 cases. This hybrid approach improves the prediction accuracy and provides actionable insights by leveraging both unstructured and structured data.

which can predict 'Used Technologies' for new test data

- 6) After training we have evaluate random forest performance in terms of accuracy and confusion matrix

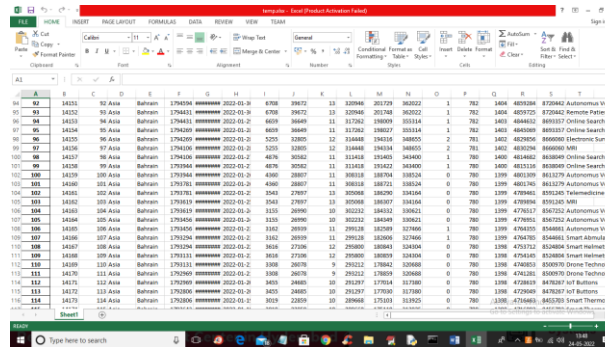
SCREEN SHOTS

Below are the output screen shots which we done in JUPYTER NOTEBOOK Python

In each output screen we have wrote comments starting with # symbol in light blue colour text

Results

In this project as per your instructions we have added 'Used Technologies' label to entire dataset and below screen showing that output

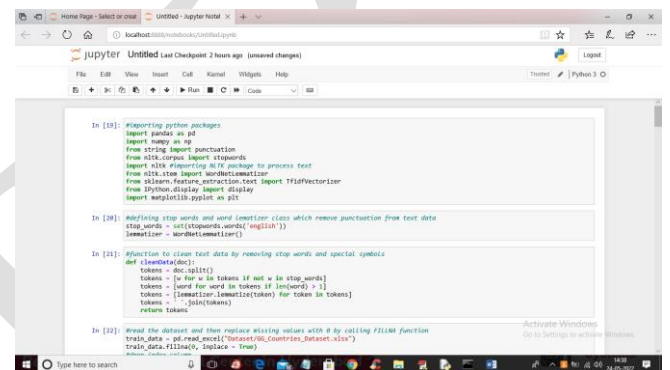


In above screen in last column you can see we have added 'Technologies Used' label in all columns.

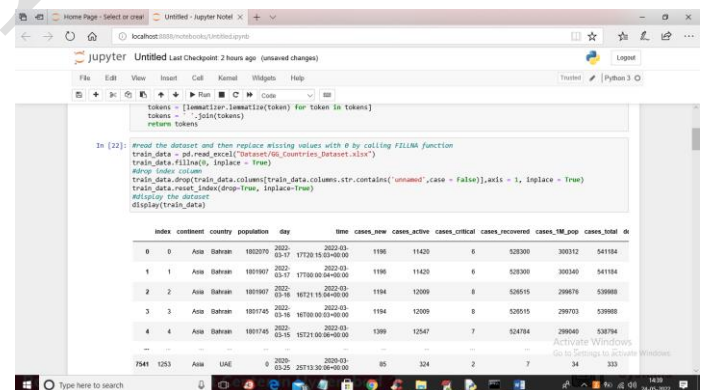
In this project we have used NLTK (natural language toolkit) for text analysis to process text data and then convert this text data into numeric vector called TF-IDF (term frequency inverse document frequency) which will replace all TEXT data with its average frequency.

After performing text into numeric vector we have perform numeric analysis such as

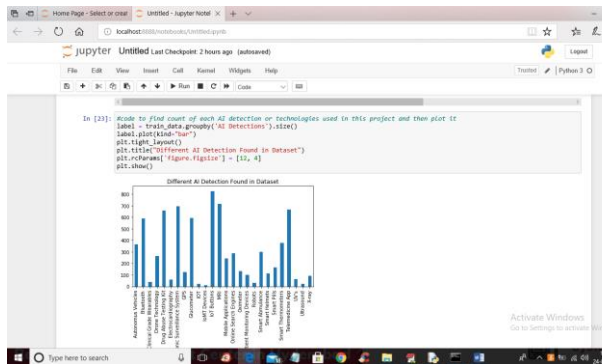
- 1) Finding count of each AI detection or technologies used
- 2) Finding sum of all critical cases country wise
- 3) Finding sum of all active cases country wise
- 4) Finding sum of all NEW cases country wise
- 5) Split dataset into train and test and then apply machine learning algorithm called Random Forest to build an AI model



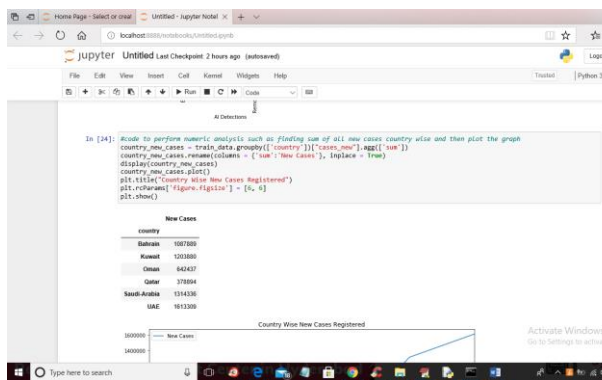
In above screen we are loading python classes and packages and then defining function to clean data and then defining stop words and word lematizers object



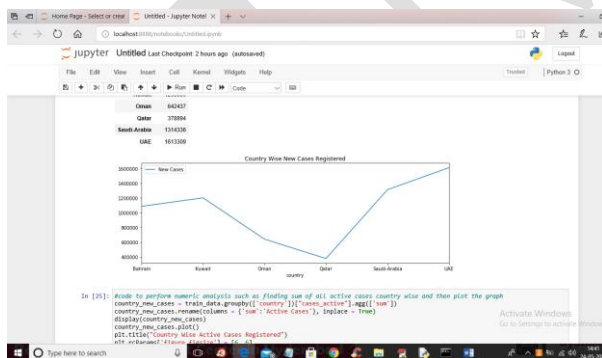
In above screen we are reading and displaying dataset



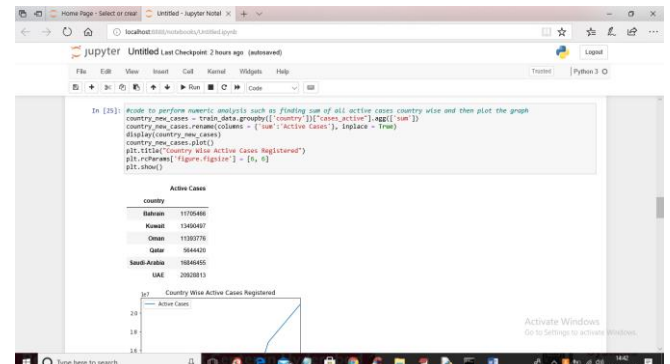
In above screen we are finding count of each USED technologies and then plotting graph where x-axis contains technology name and y-axis contains count of that technology found in dataset



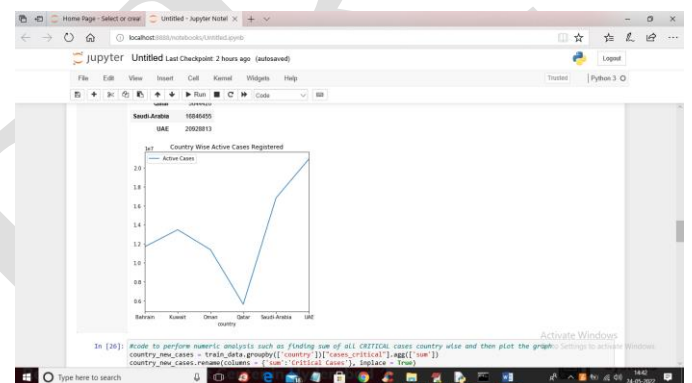
In above screen we are performing numeric analysis such as finding total NEW CASES country and then plotting graph



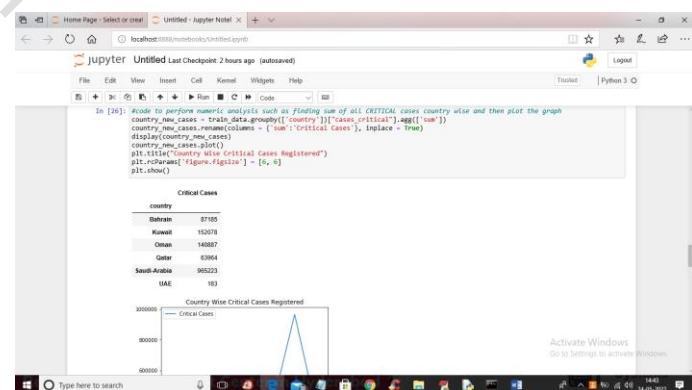
In above graph x-axis contains country name and y-axis contains total NEW CASES found in those countries and similarly we find ACTIVE & CRITICAL cases

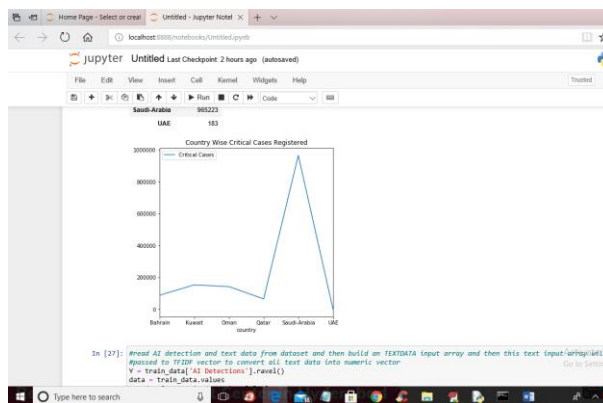


Above screen showing analysis of ACTIVE cases and below is the graph



In above graph we can see country wise active cases and below code showing analysis for CRITICAL cases





In above graph we can see critical cases analysis and in below screen we are performing TEXT analysis to read all text data and then convert to numeric vector

```

In [27]: Read AI detection and text data from dataset and then build on TEXTDATA input array and then this input array will
         be used to TFIDF vector to convert all text data into numeric vector
         y = train_data[AI_Detections].ravel()
         data = train_data.values
         X = data[:,train_data.shape[1]-1]
         textdata = []
         for i in range(len(X)):
             words = str(X[i]).replace(",","")
             words = words.lower()
             textdata.append(words)
         label_data = np.unique(y).tolist()
         print("Various AI Detections found in dataset : \n\n"+str(label_data)+"\n\n")

Various AI Detections found in dataset :
['Autonomous Vehicles', 'Bluetooth', 'Clinical Grade Wearables', 'Drone Technology', 'Drug Abuse Testing Kit', 'Electronic
phy', 'Electronic Surveillance System', 'GPS', 'Glucometer', 'IoT', 'IoT Devices', 'IoT Buttons', 'IoT', 'Mobile Appli
cations', 'Online Search Engines', 'Outlets', 'Remote Patient Monitoring Devices', 'Robots', 'Smart Bandwatches', 'Smart Home
art Pills', 'Smart Thermometers', 'Telemedicine App', 'UV's', 'Ultrasound', 'X-ray']

In [28]: Convert text data into vector where you can see all words in first row and remaining rows contains average frequency
         of those words
         tfidf_vectorizer = TfidfVectorizer(stop_words=stop_words, use_idf=True, smooth_idf=False, norm=None, decode_error='ignore')
         df = pd.DataFrame(tfidf_vectorizer.fit_transform(textdata).toarray())
         display(df)
         df = df.values
         X = df[:,0:df.shape[1]]
    
```

In above screen we are reading entire text data and then finding names of different TECHNOLOGIES used and then displaying all those names and below screen showing VECTOR

```

In [28]: Convert text data into vector where you can see all words in first row and remaining rows contains average
         of those words
         tfidf_vectorizer = TfidfVectorizer(stop_words=stop_words, use_idf=True, smooth_idf=False, norm=None, decode_error='ignore')
         df = pd.DataFrame(tfidf_vectorizer.fit_transform(textdata).toarray())
         display(df)
         df = df.values
         X = df[:,0:df.shape[1]]

00 01 0100 0101 0106 0107 0108 0109 0110 0111 ... 99997 arabia asia bahrain kuwait oman qatar sa
0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.0 0.0 1.000795 2.46541 0.0 0.0 0.0
1 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.0 0.0 1.000795 2.46541 0.0 0.0 0.0
2 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.0 0.0 1.000795 2.46541 0.0 0.0 0.0
3 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.0 0.0 1.000795 2.46541 0.0 0.0 0.0
4 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.0 0.0 1.000795 2.46541 0.0 0.0 0.0
...
7541 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.0 0.0 1.000795 0.00000 0.0 0.0 0.0
7542 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.0 0.0 1.000795 0.00000 0.0 0.0 0.0
7543 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.0 0.0 1.000795 0.00000 0.0 0.0 0.0
7544 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 ... 0.0 0.0 1.000795 0.00000 0.0 0.0 0.0
    
```

In above screen we have converted all textual data to numeric vector and this vector can be input machine learning algorithm to train a model and

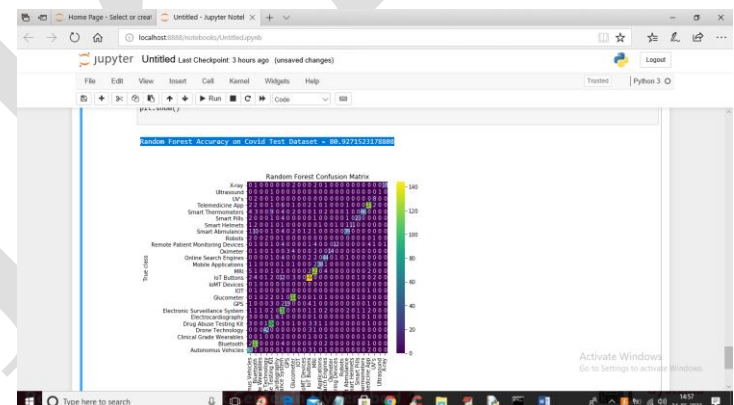
this model can be applied on NEW TEST data to predict USED TECHNOLOGIES

```

In [29]: Import machine learning packages and classes
         from sklearn.preprocessing import LabelEncoder
         from sklearn.ensemble import RandomForestClassifier
         from sklearn.metrics import accuracy_score
         from sklearn.model_selection import train_test_split
         import random as rnd
         from sklearn.metrics import confusion_matrix

         le = LabelEncoder()
         y = le.fit_transform(y)
         # Split dataset into train and test where 80% dataset used for training and 20% for testing
         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)
         # Create random forest on training data
         rf = RandomForestClassifier()
         rf.fit(X_train, y_train)
         # Predict on test data
         accuracy = accuracy_score(y_test, predict) * 100 # Calculate accuracy
         print("Random Forest Accuracy on Covid Test Dataset = "+str(accuracy)+"%")
         # Display confusion matrix of predicted used technologies
         conf_matrix = confusion_matrix(y_test, predict)
         plt.figure(figsize=(10,10))
         ax = sns.heatmap(conf_matrix, xticklabels=label_data, yticklabels=label_data, annot=True, cmap="viridis", fmt=".2g");
         plt.xlabel('True Class')
         plt.ylabel('Predicted Class')
         plt.title('Random Forest Confusion Matrix')
         plt.show()
    
```

In above screen you can see code to train Random Forest algorithm and then trained Random Forest model can be applied on TEST data and then calculate prediction accuracy and confusion matrix



In above screen in blue colour text you can see we got Random Forest accuracy as 80% and in confusion matrix graph x-axis contains PREDICTED classes and y-axis contains TRUE classes and in diagonal boxes we can see both predicted and true classes prediction are correct

Conclusion

In this project, we developed a hybrid machine learning framework to predict COVID-19 severity by combining text analysis with numerical data processing. Using NLP techniques such as tokenization, lemmatization, and TF-IDF vectorization, we effectively transformed unstructured text data into a format suitable for machine learning. The Random Forest algorithm was employed to train a predictive model, achieving 80% accuracy in identifying technologies used and assessing case severity across countries.

The integration of textual insights with numerical COVID-19 case data enabled a more holistic understanding of the pandemic's impact. Overall, the proposed approach demonstrates the potential of combining machine learning and deep learning techniques for robust and accurate health crisis prediction and management.

References

1. Arora, P., Kumar, H., & Panigrahi, B. K. (2021). Prediction and analysis of COVID-19 positive cases using deep learning models: A descriptive case study of India. *Chaos, Solitons & Fractals*, 139, 110017. <https://doi.org/10.1016/j.chaos.2020.110017>
2. Alakus, T. B., & Turkoglu, I. (2020). Comparison of deep learning approaches to predict COVID-19 infection. *Chaos, Solitons & Fractals*, 140, 110120. <https://doi.org/10.1016/j.chaos.2020.110120>
3. Oyelade, O. N., & Ezugwu, A. E. (2021). COVID-19: A natural language processing and machine learning perspective. *Soft Computing*, 25(8), 6023–6042. <https://doi.org/10.1007/s00500-021-05637-1>
4. Joshi, A., Chatterjee, S., & Kumar, A. (2021). Deep learning enabled COVID-19 detection and its severity prediction using chest X-ray images. *Biomedical Signal Processing and Control*, 70, 102960. <https://doi.org/10.1016/j.bspc.2021.102960>
5. Kumar, A., Gupta, P. K., & Srivastava, A. (2020). A review of modern technologies for tackling COVID-19 pandemic. *Diabetes & Metabolic Syndrome: Clinical Research & Reviews*, 14(4), 569–573.