**CHAPTER-1**

**INTRODUCTION**

* 1. **Introduction:**

In times of disaster events such as earthquake, flood, and hurricane, social media platforms can play a critical role in spreading a large volume of important information [1]– [3]. People frequently use these social media platforms to communicate at different hierarchies such as from individual to individual, individual to government, individual to community and government to people [3], [4]. Victim often share information about disaster events on Twitter, such as reporting about injured or deceased people, and infrastructural damages. Affected people also inquire for urgent aids by posting images, tweets, and videos. Analyzing such social media posts and extracting actionable insights in real-time can be very helpful for humanitarian organizations to assist the affected people [5], [6]. However, it is very difficult and time-consuming task to manually analyze and extract actionable insights from large amount of crisis-related tweets.

The humanitarian computing community has attempted to address the above challenge by developing automated systems that can extract and classify crisis-related social media posts [7]–[9]. These systems include classifiers to identify event types (example: flood, hurricane) [10], whether a post is informative or not [11], as well as humanitarian information types (e.g., types of damages) [12]. Despite such recent progress, existing works are primarily limited in two ways. First, most works on for damage or disaster response from social media posts have mainly concentrated on textual or image content analysis independently. However, recent studies suggest that information from both texts and images often provides valuable insights about an event and thus leads to more precise inferences than the learning from unimodal content [13]. Second, a very few works that utilize multimodal features focus on applying CNN or RNN models for text feature representation [7], [8], which might not work well for longer sentences.

In this project, our objective is to develop an effective computational model for identifying disaster-related information by integrating features from textual modality alone. Specifically, we aim to extract textual features using an attention mechanism integrated with the Bidirectional Long Short-Term Memory (BiLSTM) network to address the challenge of capturing long-range dependencies in text. We then aggregate these textual features using deep fusion techniques and apply the softmax layer to classify the given tweet. We conduct extensive experiments on a dataset focused on classifying damage types (e.g., fire, floods, infrastructure damage) from text alone. Our experiments involve comparing our model with several baselines that do not utilize multimodal features or attention mechanisms. Our key findings suggest that utilizing attention mechanisms can enhance model performance.

**1.2 The primary contributions of our project are as follows:**

1. Proposal of a unimodal architecture utilizing BiLSTM with attention mechanism to classify damage-related posts, leveraging textual information only.

2. Comparison of the proposed model's performance with existing unimodal classification techniques.

3. Empirical evaluation of the proposed model on a benchmark dataset, demonstrating the effectiveness of introducing attention mechanisms.

4. Conducting both quantitative and qualitative analysis to gain deeper insights into error types, which can inform future model improvements.

**1.3 Problem Statement:**

The increasing frequency and severity of natural disasters pose significant threats to human safety and well-being. Social media platforms have emerged as valuable sources of real-time information during such events, with users sharing crucial data through various modalities such as text, images, and videos. However, the sheer volume and diversity of information make it challenging to identify and prioritize relevant posts for effective disaster management.

Existing approaches to disaster identification from social media posts often struggle to provide accurate and timely insights due to their limited capacity to understand and integrate information from multiple modalities. The lack of a robust and deep learning model capable of effectively capturing the contextual nuances and correlations within text hinders the development of efficient disaster identification systems.

Therefore, there is a pressing need for a novel unimodal approach that leverages deep attentive learning techniques to enhance the accuracy and efficiency of disaster identification from text-based social media posts. This project should address the challenges of processing large-scale, unstructured data while effectively integrating textual information. Developing such a comprehensive model holds the key to advancing the capabilities of disaster response teams, enabling them to make more informed decisions and allocate resources more effectively in the face of natural calamities.

**1.4 Scope:**

**1.Data Preprocessing:**

* **Text Cleaning and Normalization:** Remove irrelevant characters, URLs, and stop words.
* **Tokenization:** Split text into individual words or sub-words.
* **Word Embedding:** Use Word2Vec or GloVe to represent words as dense vectors, capturing semantic relationships.

**2.Feature Extraction:**

**BiLSTM Network:**

* Captures long-range dependencies in tweet text.
* Enhances contextual understanding.

**Attention Mechanism:**

* + - Focuses on relevant words for disaster identification.
    - Improves key information extraction.

**Methodology:**

* + - Utilizes BiLSTM and attention mechanism.
    - Enhances disaster-related information identification.

**3.Classification:**

* Utilizes softmax classifier for tweet categorization by damage types.
* Example: fire, floods, infrastructure damage.

**4.Model Training and Evaluation:**

* The model is trained on a labeled dataset of disaster-related tweets.
* Performance is evaluated using metrics such as accuracy, precision, recall, and F1-score.

**1.5 Objective:**

**1. Develop a Deep Attentive Learning Model:**

Design and implement a sophisticated deep learning architecture capable of effectively processing and integrating textual information from social media posts.

**2. Enhance Textual Understanding:**

Implement natural language processing (NLP) techniques to extract relevant information from text-based content in social media posts, considering linguistic nuances and context for accurate disaster identification.

**3. Integrate Attention Mechanisms:**

Incorporate attention mechanisms within the model to dynamically weight and prioritize different segments of social media content, enhancing the focus on critical information during disaster identification.

**4. Address Fusion Challenges:**

Tackle the complexities of fusing information from diverse textual sources by exploring and implementing effective strategies for combining textual representations, ensuring a comprehensive understanding of the content.

**5. Enable Real-Time Processing:**

Optimize the deep attentive learning model for real-time processing, allowing for swift identification and classification of disaster-related content as it emerges on social media platforms.

**CHAPTER-2**

**LITERATURE SURVEY**

**2.1 K. K. Kapoor, K. Tamilmani, N. P. Rana, P. Patil, Y. K. Dwivedi, and S. Nerur, “Advances in social media research: Past, present and future,” Information Systems Frontiers, vol. 20, no. 3, pp. 531–558, 2018.**

Social media comprises communication websites that facilitate relationship forming between users from diverse backgrounds, resulting in a rich social structure. User generated content encourages inquiry and decision-making. Given the relevance of social media to various stakeholders, it has received significant attention from researchers of various fields, including information systems. There exists no comprehensive review that integrates and synthesises the findings of literature on social media. This study discusses the findings of 132 papers (in selected IS journals) on social media and social networking published between 1997 and 2017. Most papers reviewed here examine the behavioural side of social media, investigate the aspect of reviews and recommendations, and study its integration for organizational purposes. Furthermore, many studies have investigated the viability of online communities/social media as a marketing medium, while others have explored various aspects of social media, including the risks associated with its use, the value that it creates, and the negative stigma attached to it within workplaces. The use of social media for information sharing during critical events as well as for seeking and/or rendering help has also been investigated in prior research. Other contexts include political and public administration, and the comparison between traditional and social media. Overall, our study identifies multiple emergent themes in the existing corpus, thereby furthering our understanding of advances in social media research. The integrated view of the extant literature that our study presents can help avoid duplication by future researchers, whilst offering fruitful lines of enquiry to help shape research for this emerging field.

**2.2 J. Kim and M. Hastak, “Social network analysis: Characteristics of online social networks after a disaster,” International Journal of Information Management, vol. 38, no. 1, pp. 86–96, 2018.**

Social media, such as Twitter and Facebook, plays a critical role in disaster management by propagating emergency information to a disaster-affected community. It ranks as the fourth most popular source for accessing emergency information. Many studies have explored social media data to understand the networks and extract critical information to develop a pre- and post-disaster mitigation plan.

The 2016 flood in Louisiana damaged more than 60,000 homes and was the worst U.S. disaster after Hurricane Sandy in 2012. Parishes in Louisiana actively used their social media to share information with the disaster-affected community − e.g., flood inundation map, locations of emergency shelters, medical services, and debris removal operation. This study applies social network analysis to convert emergency social network data into knowledge. We explore patterns created by the aggregated interactions of online users on Facebook during disaster responses. It provides insights to understand the critical role of social media use for emergency information propagation. The study results show social networks consist of three entities: individuals, emergency agencies, and organizations. The core of a social network consists of numerous individuals. They are actively engaged to share information, communicate with the city of Baton Rouge, and update information. Emergency agencies and organizations are on the periphery of the social network, connecting a community with other communities. The results of this study will help emergency agencies develop their social media operation strategies for a disaster mitigation plan.

**2.3 J. Son, H. K. Lee, S. Jin, and J. Lee, “Content features of tweets for effective communication during disasters: A media synchronicity theory perspective,” International Journal of Information Management, vol. 45, pp. 56–68, 2019.**

Users’ ability to retweet information has made Twitter one of the most prominent social media platforms for disseminating emergency information during disasters. However, few studies have examined how Twitter’s features can support the different communication patterns that occur during different phases of disaster events. Based on the literature of disaster communication and Media Synchronicity Theory, we identify distinct disaster phases and the two communication types—crisis communication and risk communication—that occur during those phases. We investigate how Twitter’s representational features, including words, URLs, hashtags, and hashtag importance, influence the average retweet time—that is, the average time it takes for retweet to occur—as well as how such effects differ depending on the type of disaster communication. Our analysis of tweets from the 2013 Colorado floods found that adding more URLs to tweets increases the average retweet time more in risk-related tweets than it does in crisis-related tweets. Further, including key disaster-related hashtags in tweets contributed to faster retweets in crisis-related tweets than in risk-related tweets. Our findings suggest that the influence of Twitter’s media capabilities on rapid tweet propagation during disasters may differ based on the communication processes.

**2.4 A. Elbanna, D. Bunker, L. Levine, and A. Sleigh, “Emergency management in the changing world of social media: Framing the research agenda with the stakeholders through engaged scholarship,” International Journal of Information Management, vol. 47, pp. 112–120, 2019.**

The use of social media and Web 2.0 platforms is proliferating and affecting different formal and highly structured organizations including public safety agencies. Much of the research in the area has focused on public use of social media during an emergency as well as how emergency agencies benefit from the data and information generated by this process. However, there is little understanding of “what are the operational implications of this public use on emergency management agencies and how does social media either positively or negatively impact these operations”? In order to progress research into this topic, we chose an engaged scholarship framework to shape a research agenda with the active participation of stakeholders. Hence, we conducted a series of workshops primarily involving over 100 public safety practitioners working in the area of disasters and emergency management who work in public safety agencies, humanitarian organizations, volunteering online platforms and volunteer groups in addition to 20 academics working on this area of enquiry. The findings highlight six different challenges that emergency responding organizations currently face in relation to social media use. We conceptualize these challenges as creating six operational tension zones for organizations. We discuss these tensions and their implications for future research and practice.

**2.5 R. Dubey, A. Gunasekaran, S. J. Childe, D. Roubaud, S. F. Wamba, M. Giannakis, and C. Foropon, “Big data analytics and organizational culture as complements to swift trust and collaborative performance in the humanitarian supply chain,” International Journal of Production Economics, vol. 210, pp. 120–136, 2019.**

The main objective of the study is to understand how big data analytics capability (BDAC) as an organizational culture can enhance trust and collaborative performance between civil and military organizations engaged in disaster relief operations. The theoretical framework is grounded in organizational information processing theory (OIPT). We have conceptualized an original theoretical model to show, using the competing value model (CVM), how BDAC, under a moderating influence of organizational culture, affects swift trust (ST) and collaborative performance (CP). We used WarpPLS 6.0 to test the proposed research hypotheses using multi-respondent data gathered through an email questionnaire sent to managers working in 373 organizations, including the military forces of different countries, government aid agencies, UN specialized agencies, international non-government organizations (NGOs), service providers, and contractors. The results offer four important implications. First, BDAC has a positive, significant effect on ST and CP. Second, flexible orientation (FO) and controlled orientation (CO) have no significant influence on building ST. Third, FO has a positive and significant moderating effect on the path joining BDAC and CP. Finally, CO has negative and significant moderating effect on the path joining BDAC and CP. The control variables: temporal orientation (TO) and interdependency (I) have significant effects on ST and CP. These results extend OIPT to create a better understanding of the application of information processing capabilities to build swift trust and improve collaborative performance. Furthermore, managers can derive multiple insights from this theoretically-grounded study to understand how BDAC can be exploited to gain insights in contexts of different management styles and cultures. We have also outlined the study limitations and provided numerous future research directions.

**CHAPTER-3**

**SYSTEM ANALYSIS**

**3.1 EXISTING SYSTEM:**

* Aipe et al. [22] also proposed a CNN-based approach but they focus on multi label classification rather that simple binary classification to label disaster-related tweets. Similarly, Yu et al. [23] used CNN, logistic regression, and SVM to classify the tweets related to different Hurricanes into multiple categories. Their CNN-based model outperformed SVM and LR. In contrast to CNN-based approaches we consider BiLSTMs with attention mechanisms with an aim to better captures dependencies between word tokens.
* Li et al. [24] studied the feasibility of domain adaption for analyzing the disaster tweets by applying the naive Bayes classifier on the Boston Marathon bombing and Hurricane Sandy dataset. Graf et al. [25] focused on cross-domain classification so that the classifier can be used across different types disaster events. They employed a cross-domain classifier and utilized emotional, sentimental, and linguistic features extracted from the damage-related tweets. Others have focused on text mining and summarization approaches [26], [27]. For example, Rudra et al. [26] assign tweets into different situational classes and then summarizes those tweets. Cameron et al. [27] proposed an Emergency Situation Awareness-Automated Web Text Mining (ESAAWTM) system that detects informative damage-related Twitter messages to inform charitable organizations about the incidents of a disaster. Unlike these systems that broadly focused on text mining and summarization, we only focus specifically on a multi-class classification problem on disaster-related tweets.
* Nguyen et al. [29] developed a deep CNN architecture to label the social media images into multiple disaster categories (i.e., severe, mild, and no-damage). Similarly, Alam et al. [30] proposed a pretrained CNN (VGG16) based framework that can identify the disaster images uploaded on the online platforms. Daly and Thom [31] culled flicker images to detect the fire event using pretrained classifiers. Finally, Lagerstrom et al. [32] developed a system to classify whether the image indicates a fire event or not. In contrast to these works that broadly developed binary classifier for classifying disaster vs.
* Chen et al. [34] studied the relation between the images and texts and utilize visual features along with socially relevant contextual features (e.g., time of posting, the number of comments, retweets) to identify disaster information. et al. [7] explored damage detection by focusing on human and environmental damage related posts. They used the Inception pre-trained model for visual feature extraction and designed a CNN architecture for textual features.
* Similarly, Rizk et al. [35] proposed a multimodal architecture to classify the Twitter data into infrastructure and natural damage categories. Ferda et al. [8] also presented a multimodal approach for classifying the tweets into two categories: informative task (e.g., informative vs. non-informative) and humanitarian task (e.g., affected individuals, rescue volunteering or donation effort, infrastructure and utility\ damage). They used CNN based approach for extracting the visual and textual features. Gautam et al. [36] showed a comparison between unimodal and multimodal methods on CrisisMMD [37] dataset. They utilized the late fusion [38] approach for combining the image-tweet pairs. All the works reported significant performance improvement using multimodal information in contrast to their counterparts that utilize uni-modal information.

**3.2 Problems in Existing System**

* Proposed a CNN-based approach but they focus on multilabel classification rather that simple binary classification to label disaster-related tweets.
* Used CNN, logistic regression, and SVM to classify the tweets related to different Hurricanes into multiple categories.

**3.3 Proposed System:**

* The primary contributions of our project are: We propose a unimodal architecture that utilizes BiLSTM recurrent neural network with attention mechanism to classify the damage-related posts by exploiting textual information.
* We compare the performance of the proposed model with a set of existing unimodal (i.e., image, text) and multimodal classification techniques.
* We empirically evaluate the proposed model on a benchmark dataset and demonstrated how introducing attention could enhance the system performance through an intrinsic evaluation.
* We perform both quantitative and qualitative analysis to get deeper insights about the error types which provide future directions for improving the model.

**3*.*4 Advantages of Proposed system:**

* In the proposed project, the system develops an effective computational model for identifying disaster-related information by integrating features textual modalities.
* In the proposed system, the system transforms the tweet into a vector representation and then use an embedding layer to obtain semantic representations (embedding features) of the words.

**3.5 Project Requirements:**

The project involved analyzing the design of few applications so as to make the application more users friendly. To do so, it was really important to keep the navigations from one screen to the other well ordered and at the same time reducing the amount of typing the user needs to do. In order to make the application more accessible, the browser version had to be chosen so that it is compatible with most of the Browsers.

**1.Functional Requirements**

* Graphical User interface with the User.

**2.Non-Functional Requirements**

**Maintainability:** Maintainability is used to make future maintenance easier, meet new requirements. Our project can support expansion.

**Robustness:** Robustness is the quality of being able to withstand stress, pressures or changes in procedure or circumstance. Our project also provides it.

**Reliability:** Reliability is an ability of a person or system to perform and maintain its functions in circumstances. Our project also provides it.

**Size:** The size of a particular application plays a major role, if the size is less then efficiency will be high. The size of database we have developed is 5.05 MB.

**Speed:** If the speed is high then it is good. Since the no of lines in our code is less, hence the speed is high.

**Power Consumption:** In battery-powered systems, power consumption is very important. In the requirement stage, power can be specified in terms of battery life.

However, the allowable wattage can’t be defined by the customer. Since the no of lines of code is less CPU uses less time to execute hence power usage will be less.

**3. 6 SYSTEM REQUIREMENTS:**

**H/W System Configuration:**

➢ Processor : Pentium –IV

➢ RAM : 4 GB (min)

➢ Hard Disk : 20 GB

➢ Key Board : Standard Windows Keyboard

➢ Mouse : Two or Three Button Mouse

➢ Monitor : SVGA

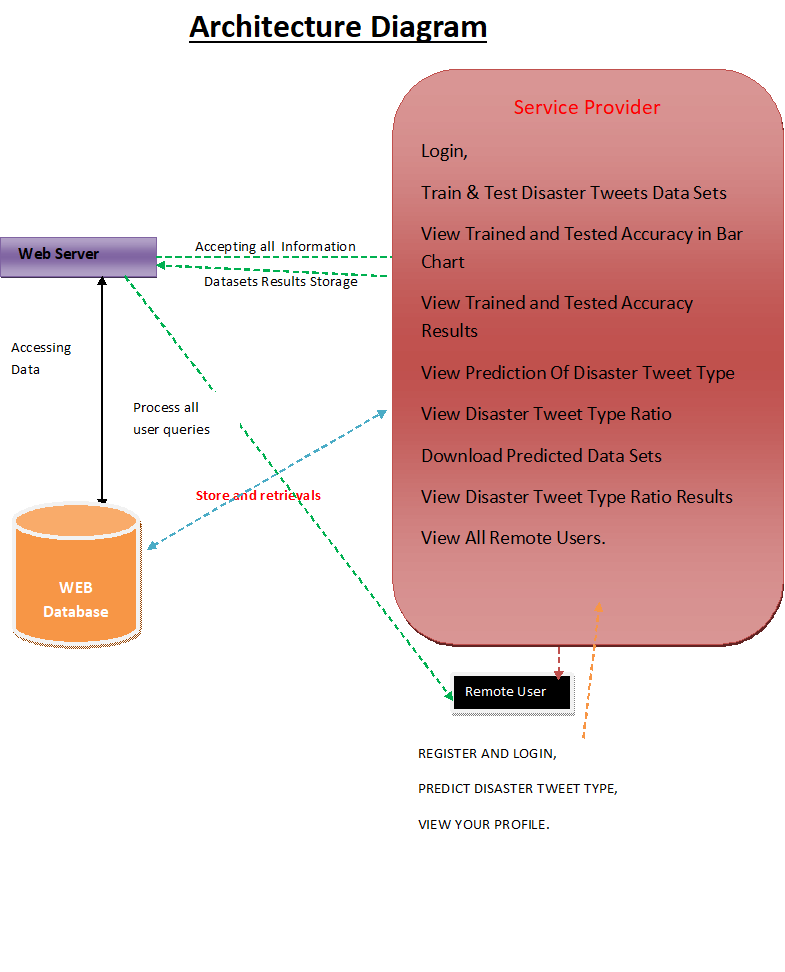
**Software Requirements:**

* Operating system : Windows 7 Ultimate.
* Coding Language : Python.
* Back-End : Django-ORM
* Designing : Html, css, javascript.
* Data Base : MySQL (WAMP Server).

**CHAPTER – 4**

**SYSTEM DESIGN**

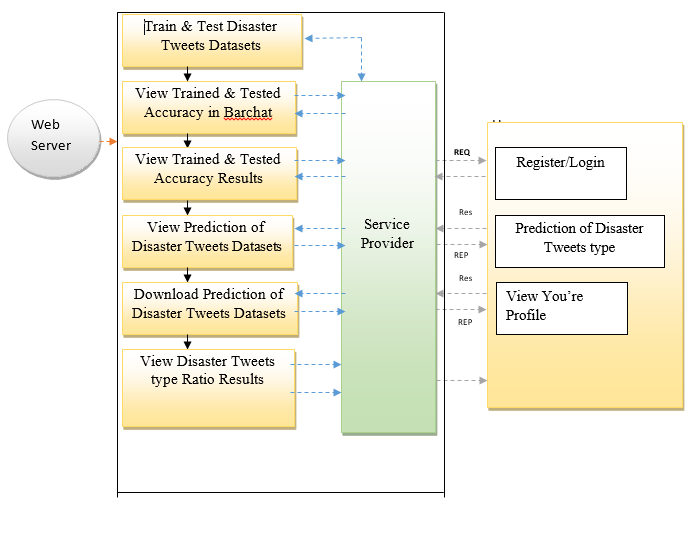
**4.1 System Architecture:**



**Figure 4.1: Architecture Diagram**

**4.2 DETAILED DESIGN:**

**4.2.1 BLOCK DIAGRAM**



**Figure 4.2.1: Block Diagram**

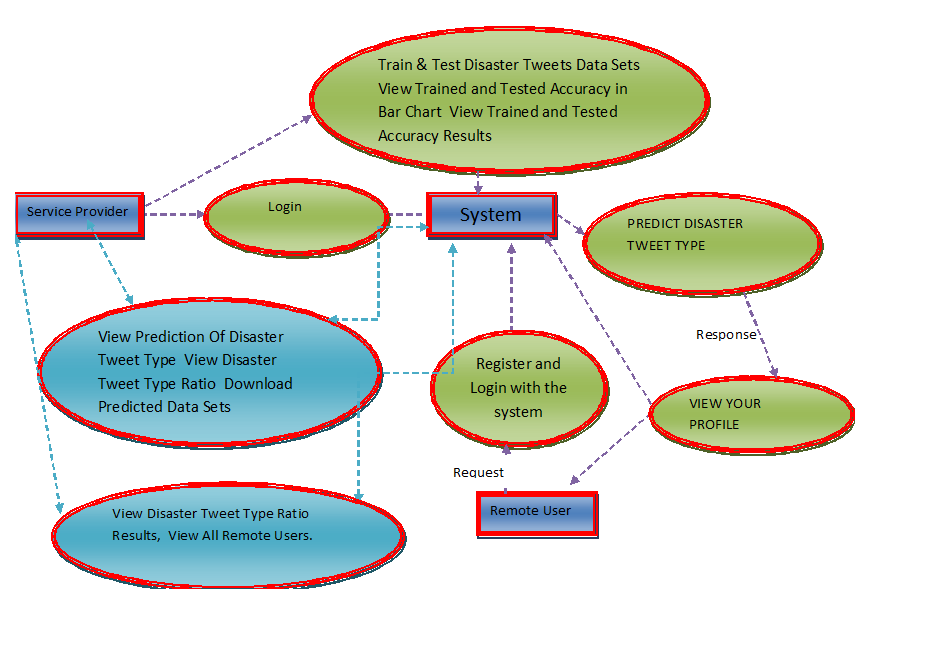
**4.2.2 DATA FLOW DIAGRAM:**

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.

2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.

3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.

4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail



**Figure 4.2.2: Data Flow Diagram**

**4.3 UML Diagrams:**

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

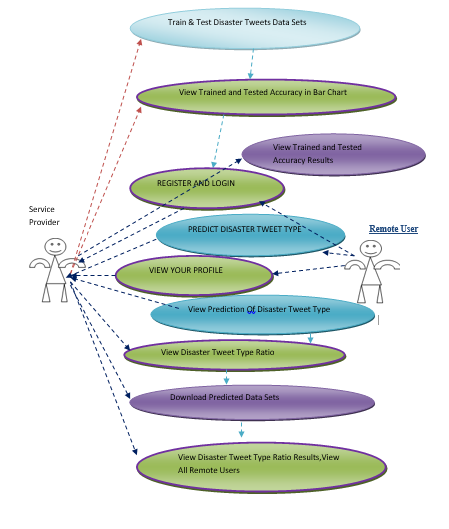
**GOALS:**

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

**4.3.1 USE CASE DIAGRAM:**

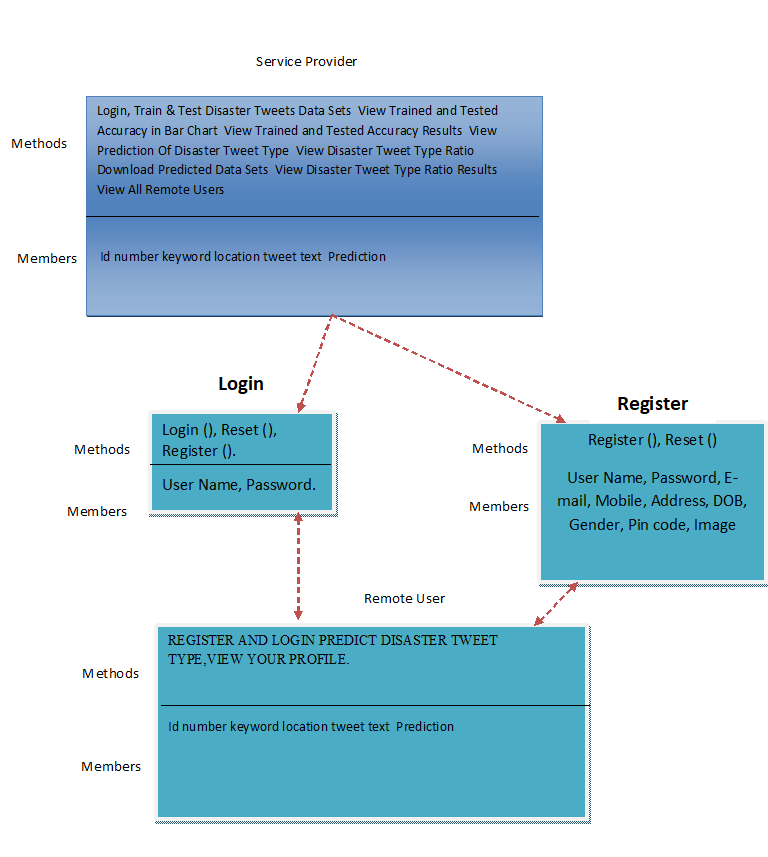
A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



**Figure 4.3.1: Use Case Diagram**

**4.3.2 CLASS DIAGRAM:**

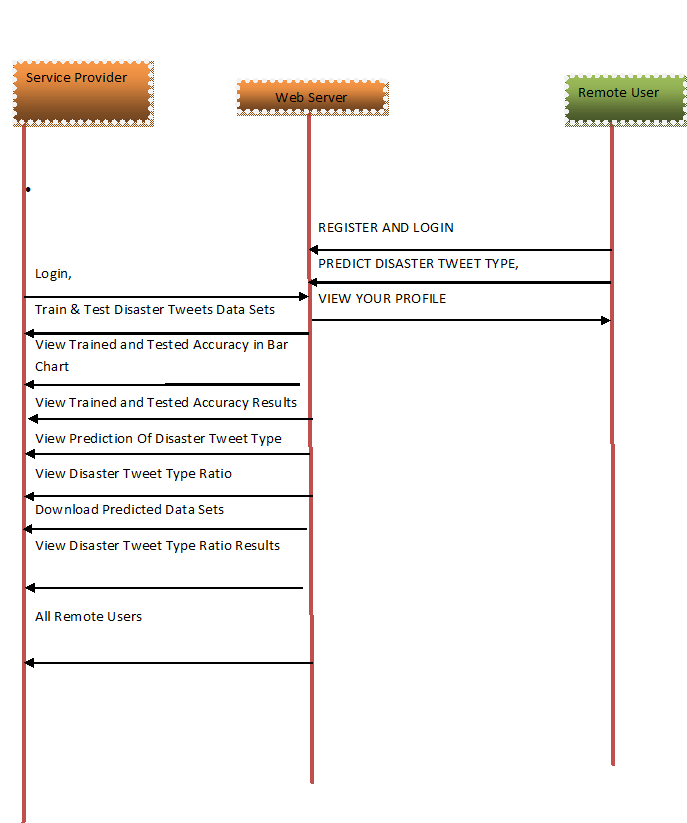
In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



**Figure 4.3.2: Class Diagram**

**4.3.3 SEQUENCE DIAGRAM:**

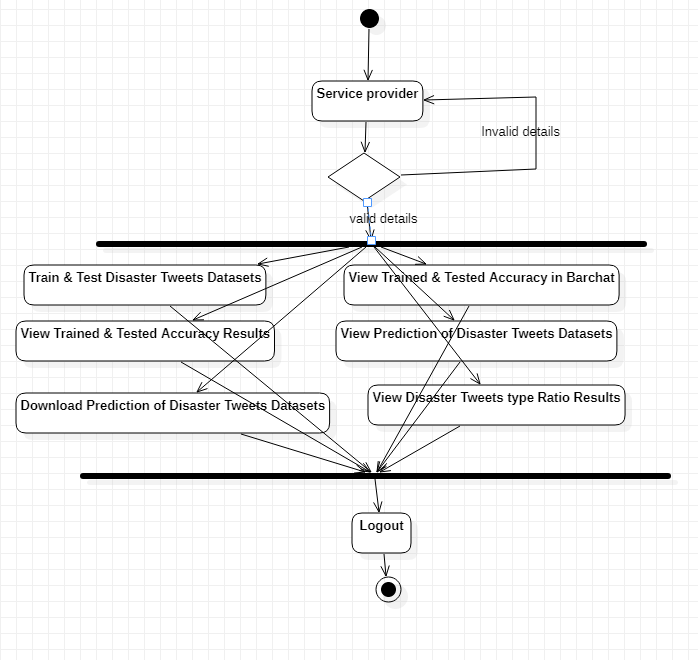
A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



**Figure 4.3.3: Sequence Diagram**

**4.3.4 ACTIVITY DIAGRAM:**

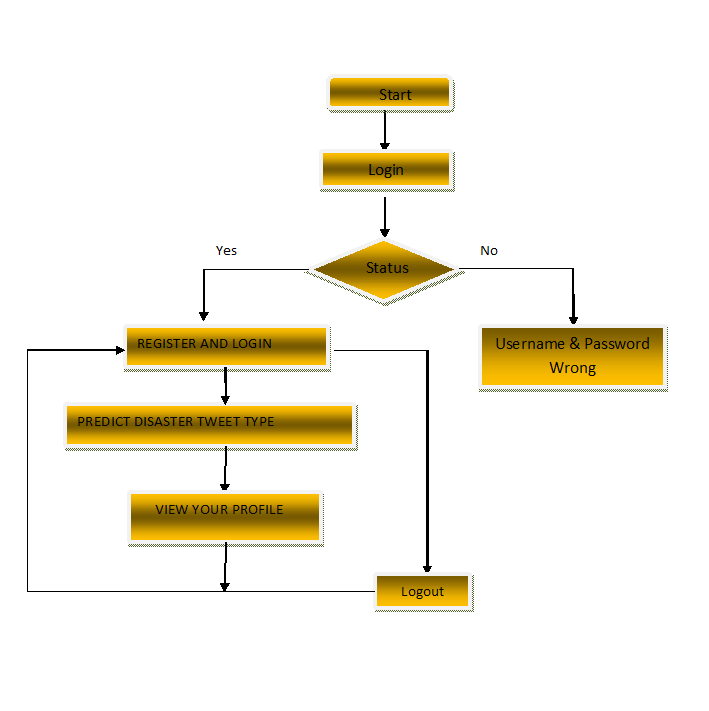
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



**Figure 4.3.4: Activity Diagram**

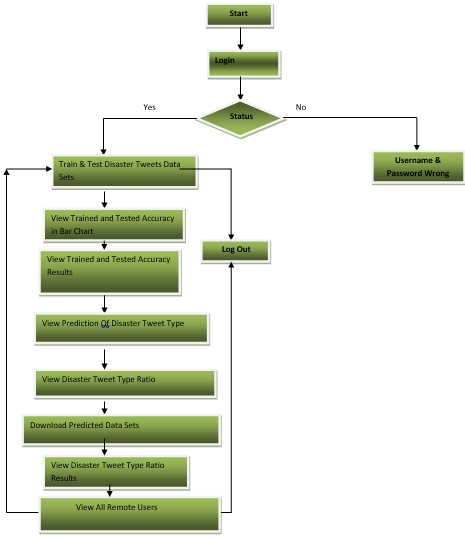
* + 1. **FLOW CHAT DIAGRAM:**

**4.4.1 Flow Chart: Remote User**



**Figure 4.4.1: Flow Chart of Remote User**

* + - 1. **Flow Chart:** Service Provider



**Figure 4.4.2: Flow Chart of Service Provider**

**CHAPTER – 5**

**MODULES**

**5.1 Service Provider:**

In this module, the Service Provider has to login by using valid user name and password. After login successful he can do some operations such as Login, Train & Test Disaster Tweets Data Sets ,View Trained and Tested Accuracy in Bar Chart, View Trained and Tested Accuracy Results, View Prediction Of Disaster Tweet Type, View Disaster Tweet Type Ratio, Download Predicted Data Sets, View Disaster Tweet Type Ratio Results, View All Remote Users.

**5.2 View and Authorize Users:**

In this module, the admin can view the list of users who all registered. In this, the admin can view the user’s details such as, user name, email, address and admin authorizes the users.

**5.3 Remote User:**

In this module, there are n numbers of users are present. User should register before doing any operations. Once user registers, their details will be stored to the database. After registration successful, he has to login by using authorized user name and password. Once Login is successful user will do some operations like REGISTER AND LOGIN, PREDICT DISASTER TWEET TYPE, VIEW YOUR PROFILE.

**CHAPTER – 6**

**TECHNOLOGY DESCRIPTION**

**6.1 PYTHON:**

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. An [interpreted language](https://en.wikipedia.org/wiki/Interpreted_language), Python has a design philosophy that emphasizes code [readability](https://en.wikipedia.org/wiki/Readability) (notably using [whitespace](https://en.wikipedia.org/wiki/Whitespace_character) indentation to delimit [code blocks](https://en.wikipedia.org/wiki/Code_block) rather than curly brackets or keywords), and a syntax that allows programmers to express concepts in fewer [lines of code](https://en.wikipedia.org/wiki/Source_lines_of_code) than might be used in languages such as [C++](https://en.wikipedia.org/wiki/C%2B%2B)or [Java](https://en.wikipedia.org/wiki/Java_(programming_language)). It provides constructs that enable clear programming on both small and large scales. Python interpreters are available for many [operating systems](https://en.wikipedia.org/wiki/Operating_system). [CPython](https://en.wikipedia.org/wiki/CPython" \o "CPython), the [reference implementation](https://en.wikipedia.org/wiki/Reference_implementation) of Python, is [open source](https://en.wikipedia.org/wiki/Open_source) software and has a community-based development model, as do nearly all of its variant implementations. CPython is managed by the non-profit [Python Software Foundation](https://en.wikipedia.org/wiki/Python_Software_Foundation). Python features a [dynamic type](https://en.wikipedia.org/wiki/Dynamic_type) system and automatic [memory management](https://en.wikipedia.org/wiki/Memory_management). It supports multiple [programming paradigms](https://en.wikipedia.org/wiki/Programming_paradigm), including [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming), [imperative](https://en.wikipedia.org/wiki/Imperative_programming), [functional](https://en.wikipedia.org/wiki/Functional_programming) and [procedural](https://en.wikipedia.org/wiki/Procedural_programming), and has a large and comprehensive [standard library](https://en.wikipedia.org/wiki/Standard_library)

## **What is Python**

## **Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.**

**It is used for:**

* web development (server-side),
* software development,
* mathematics,
* system scripting.

### **What can Python do**

* Python can be used on a server to create web applications.
* Python can be used alongside software to create workflows.
* Python can connect to database systems. It can also read and modify files.
* Python can be used to handle big data and perform complex mathematics.
* Python can be used for rapid prototyping, or for production-ready software development.

### **Why Python**

* Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
* Python has a simple syntax similar to the English language.
* Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
* Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
* Python can be treated in a procedural way, an object-orientated way or a functional way.

### **Good to know**

The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.

* In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

### **Python Syntax compared to other programming languages**

* Python was designed for readability, and has some similarities to the English language with influence from mathematics.
* Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
* Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

**6.2 Introduction**

Python applications will often use packages and modules that don’t come as part of the standard library. Applications will sometimes need a specific version of a library, because the application may require that a particular bug has been fixed or the application may be written using an obsolete version of the library’s interface.

This means it may not be possible for one Python installation to meet the requirements of every application. If application A needs version 1.0 of a particular module but application B needs version 2.0, then the requirements are in conflict and installing either version 1.0 or 2.0 will leave one application unable to run.

The solution for this problem is to create a virtual environment, a self-contained directory tree that contains a Python installation for a particular version of Python, plus a number of additional packages.

Different applications can then use different virtual environments. To resolve the earlier example of conflicting requirements, application A can have its own virtual environment with version 1.0 installed while application B has another virtual environment with version 2.0. If application B requires a library be upgraded to version 3.0, this will not affect application A’s environment.

Python is the most widely used multi-purpose, high-level programming language at the moment. Python supports both Object-Oriented and Procedural programming paradigms. Python programmes are typically smaller than those written in other programming languages such as Java. Programmers must type relatively little, and the language's indentation requirement ensures that their code is always readable. Python is used by almost all tech giants, including Google, Amazon, Facebook, Instagram, Dropbox, Uber, and others. Python's greatest strength is its vast collection of standard libraries, which can be used for the following.

• Machine Learning

• GUI Applications (like Kivy, Tkinter, PyQt etc. )

• Web frameworks like Django (used by YouTube, Instagram, Dropbox)

• Image processing (like OpenCV, Pillow)

• Web scraping (like Scrapy, BeautifulSoup, Selenium)

• Test frameworks

• Multimedia

**History of Python:**

What are the similarities between Python and the alphabet? Yes, both begin with ABC. It is abundantly clear that the programming language ABC is being referred to when we talk about ABC in the context of Python. ABC is a broadly useful programming language and programming climate, which had been created in the Netherlands, Amsterdam, at the CWI (Centrum Wiskunde and Informatica). Influencing the development of Python was ABC's greatest accomplishment. Python was first thought of at the end of the 1980s. During that time, Guido van Rossum worked on a distributed operating system called Amoeba at the CWI. Guido van Rossum stated in an interview with Bill Venners 1: At Centrum voor Wiskunde en Informatica (CWI), I worked as an implementer on a team developing a language called ABC in the early 1980s. I don't have any idea how well individuals know ABC's effect on Python. I attempt to make reference to ABC's impact since I'm obliged to all that I mastered during that undertaking and to individuals who chipped away at it." Guido van Rossum went on later in the same interview: " I recalled all my experience and a portion of my dissatisfaction with ABC.

I decided to try to create a straightforward scripting language with some of ABC's best features without its drawbacks. So I began composing. I made a straightforward virtual machine, a basic parser, and a basic runtime. I adapted the various ABC parts I liked into my own creation. I developed a basic syntax, substituted indentation for curly braces or begin-end blocks for statement grouping, and a small number of powerful data types: a hash table (or word reference, as we call it), a rundown, strings, and numbers."

**How to Install Python on Windows and Mac:**

There have been several updates in the Python version over the years. The question is how to install Python? It might be confusing for the beginner who is willing to start learning Python but this tutorial will solve your query. The latest or the newest version of Python is version 3.7.4 or in other words, it is Python 3.

Note: The python version 3.7.4 cannot be used on Windows XP or earlier devices.

Before you start with the installation process of Python. First, you need to know about your System Requirements. Based on your system type i.e. operating system and based processor, you must download the python version. My system type is a Windows 64-bit operating system. So the steps below are to install python version 3.7.4 on Windows 7 device or to install Python 3. Download the Python Cheatsheet here.The steps on how to install Python on Windows 10, 8 and 7 are divided into 4 parts to help understand better.

**Download the Correct version into the system:**

**Step 1:** Go to the official site to download and install python using Google Chrome or any other web browser. OR Click on the following link: <https://www.python.org>

**Fig:6.1.1 official website of python**

Now, check for the latest and the correct version for your operating system.

**Step 2:** Click on the Download Tab.

****

**Fig 6.1.2: download latest version**

**Step 3:** You can either select the Download Python for windows 3.7.4 button in Yellow Color or you can scroll further down and click on download with respective to their version. Here, we are downloading the most recent python version for windows 3.7.4

****

**Fig 6.1.3: click download on specific version**

**Step 4:** Scroll down the page until you find the Files option.

**Step 5:** Here you see a different version of python along with the operating system.



**Fig 6.1.4: different versions of python**

• To download Windows 32-bit python, you can select any one from the three options: Windows x86 embeddable zip file, Windows x86 executable installer or Windows x86 web-based installer.

• To download Windows 64-bit python, you can select any one from the three options: Windows x86-64 embeddable zip file, Windows x86-64 executable installer or Windows x86-64 web-based installer.

Here we will install Windows x86-64 web-based installer. Here your first part regarding which version of python is to be downloaded is completed. Now we move ahead with the second part in installing python i.e. Installation

**Note:** To know the changes or updates that are made in the version you can click on the Release Note Option.

**Installation of Python**

**Step 1:** Go to Download and Open the downloaded python version to carry out the installation process. 

**Fig 6.1.5: installation**

**Step 2:** Before you click on Install Now, make sure to put a tick on Add Python 3.7 to PATH.



**Fig 6.1.6: click install now**

**Step 3:** Click on Install NOW After the installation is successful. Click on Close.



**Fig 6.1.6: close after setup**

With these above three steps on python installation, you have successfully and correctly installed Python. Now is the time to verify the installation.

Note: The installation process might take a couple of minutes.

**Verify the Python Installation:**

**Step 1:** Click on Start

**Step 2:** In the Windows Run Command, type “cmd”



**Fig 6.1.8:command prompt**

**Step 3:** Open the Command prompt option.

**Step 4:** Let us test whether the python is correctly installed. Type python –V and press Enter.



**Fig 6.1.9: check python version**

**Step 5:** You will get the answer as 3.7.4

Note: If you have any of the earlier versions of Python already installed. You must first uninstall the earlier version and then install the new one.

**Check how the Python IDLE works**

**Step 1**: Click on Start

**Step 2:** In the Windows Run command, type “python idle”



**Fig 6.1.10: python idle**

**Step 3:** Click on IDLE (Python 3.7 64-bit) and launch the program

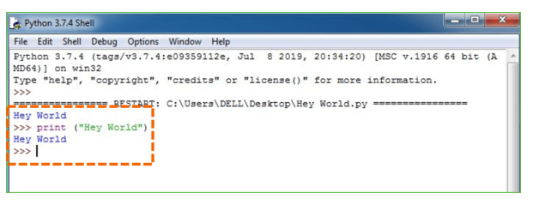
**Step 4:** To go ahead with working in IDLE you must first save the file. Click on File > Click on Save



**Fig 6.1.11 save the file**

**Step 5**: Name the file and save as type should be Python files. Click on SAVE. Here I have named the files as Hey World.

**Step 6:** Now for e.g. enter print (“Hey World”) and Press Enter.



**Fig 6.1.12: example for writing and executing**

You will see that the command given is launched. With this, we end our tutorial on how to install Python. You have learned how to download python for windows into your respective operating system.

Note: Unlike Java, Python doesn’t need semicolons at the end of the statements otherwise it won’t work.

**6.2 MACHINE LEARNING:**

Before we investigate the subtleties of different AI techniques, how about we start by seeing what AI is, and what it isn't. Although machine learning is frequently referred to as a subfield of artificial intelligence, I believe that categorization can frequently be initially misleading. The investigation of AI positively emerged from research in this specific circumstance, however in the information science use of AI techniques, it's more useful to consider AI for the purpose of building models of information.

In its most basic form, machine learning is the creation of mathematical models that aid in data comprehension. Learning" enters the conflict when we give these models tunable boundaries that can be adjusted to noticed information; The program can thus be regarded as "learning" from the data. These models can be used to predict and comprehend aspects of newly observed data once they have been fitted to data that has already been observed. The more philosophical discussion regarding the degree to which this kind of mathematical, model-based "learning" is comparable to the "learning" exhibited by the human brain will be left up to the reader. Understanding the issue setting in AI is crucial for utilizing these devices successfully, thus we will begin for certain general classifications of the kinds of approaches we'll examine here.

**Classes Of Machine Inclining:**

Machine learning can be broken down into two main categories at the most fundamental level: learning under supervision and unsupervised learning.

Modeling the relationship between the data's measured features and some label is one aspect of supervised learning. New, unidentified data can be labeled with the help of this model once it has been established. Regression and classification tasks are two more subcategories of this: The labels in classification are discrete categories, whereas the labels in regression are continuous quantities. In the following section, we will see examples of both kinds of supervised learning.

Unsupervised learning is often referred to as "letting the dataset speak for itself," and it involves modeling the features of a dataset without using any labels. These models incorporate undertakings like bunching and dimensionality decrease. Bunching calculations distinguish unmistakable gatherings of information, while dimensionality decrease calculations look for additional brief portrayals of the information. In the following section, we will see examples of both kinds of unsupervised learning.

**Need for AI**

Individuals, as of now, are the most keen and high level species on earth since they can think, assess and tackle complex issues. On the opposite side, computer based intelligence is still in its underlying stage and haven't outperformed human knowledge in numerous viewpoints. Then the inquiry that is the need to make machine learn? The most appropriate justification behind doing this is, "to decide, in view of information, with proficiency and scale".

Recently, associations are putting vigorously in fresher advancements like Man-made brainpower, AI and Profound Figuring out how to get the critical data from information to play out a few true errands and tackle issues. We can call it information driven choices taken by machines, especially to mechanize the cycle. Problems that cannot naturally be programmed can benefit from these data-driven decisions rather than programming logic. The truth of the matter is that we can't manage without human insight, however other perspective is that we as a whole need to take care of genuine issues with proficiency at a gigantic scope. Because of this, machine learning is required.

**Applications of Machines Learning:**

Researchers assert that we are in the golden age of AI and machine learning, with machine learning being the technology with the fastest growth rate. It is used to solve a lot of real-world complex problems that can't be solved using the traditional method. Emotion analysis, sentiment analysis, error detection and prevention, weather forecasting, stock market analysis and forecasting, speech synthesis, speech recognition, customer segmentation, object recognition, fraud detection and prevention, and product recommendation for online shoppers are just a few examples of ML's real-world applications.

1. **Terminologies of Machine Learning**

A model is a particular representation that is learned from data by using a machine learning algorithm. A model is likewise called a speculation.

• **Highlight** - A component is an individual quantifiable property of the information. A feature vector makes it simple to describe a collection of numerical features. The model receives input in the form of feature vectors. For instance, there may be characteristics like color, smell, taste, and so on that can be used to predict a fruit.

• **Target (Label):** The value that will be predicted by our model is referred to as a target variable or label. For the organic product model talked about in the component segment, the mark with each arrangement of information would be the name of the organic product like apple, orange, banana, and so on.

**• Training:** The idea is to provide a set of inputs (features) and expected outputs (labels), so that after training, a model (hypothesis) will map new data to one of the categories trained on.

**• Prediction:** When our model is ready, it can be given a set of inputs and a predicted output (a label) will be generated.

**(b) Types of Machine Learning**

• **Supervised Learning:** This involves using classification and regression models to learn from a training dataset with labeled data. Until the required level of performance is reached, this learning process continues.

• **Unsupervised Learning:** Using factor and cluster analysis models, this method uses unlabeled data to learn more about the data itself by locating its underlying structure.

• **Semi-supervised Learning:** This method uses a small amount of labeled data and unlabeled data, similar to unsupervised learning. Utilizing named information tremendously expands the learning precision and is likewise more savvy than Managed Learning.

**• Reinforcement Learning:** This method entails figuring out the best course of action through trial and error. Learning behaviors that are based on the current state and will maximize reward in the future determine the subsequent action.

**6.3 Packages and Versions**:

Asgiref 3.7.2

Contourpy 1.1.1

cycler 0.12.1

Django 3.0.4

et-xmlfile 1.1.0

fonttools 4.50.0

importlib-resources 6.3.0

joblib 1.3.2

kiwisolver 1.4.5

matplotlib 3.7.5

mysql-connector-python 8.0.24

mysqlclient 2.0.3

numpy 1.21.3

openpyxl 3.1.2

packaging 24.0

pandas 2.0.3

pillow 10.2.0

pip 20.1.1

protobuf 5.26.0

pyparsing 3.1.2

python-dateutil 2.9.0.post0

pytz 2024.1

scikit-learn 0.22.2.post1

scipy 1.10.1

seaborn 0.10.0

setuptools 47.1.0

six 1.16.0

sqlparse 0.4.4

typing-extensions 4.10.0

tzdata 2024.1

xlwt 1.3.0

zipp 3.18.1

**CHAPTER – 7**

**SYSTEM TESTING**

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable miner. There are various types of tests. Each test type addresses a specific testing requirement.

### **TYPES OF TESTS**

**7.1 Unit testing:**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases

**7.2 Integration testing:**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**7.3 Functional test:**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input: identified classes of valid input must be accepted.

Invalid Input: identified classes of invalid input must be rejected.

Functions: identified functions must be exercised.

Output: identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

**7.4 System Test:**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

**7.5 White Box Testing:**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cKNNot be reached from a black box level.

**7.6 Black Box Testing:**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cKNNot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

**7.7 Test strategy and approach:**

Field testing will be performed manually and functional tests will be written in detail.

**7.7.1 Test objectives**

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

**7.7.2 Features to be tested**

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**7.8 Acceptance Testing:**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

**7.9 TEST CASES:**

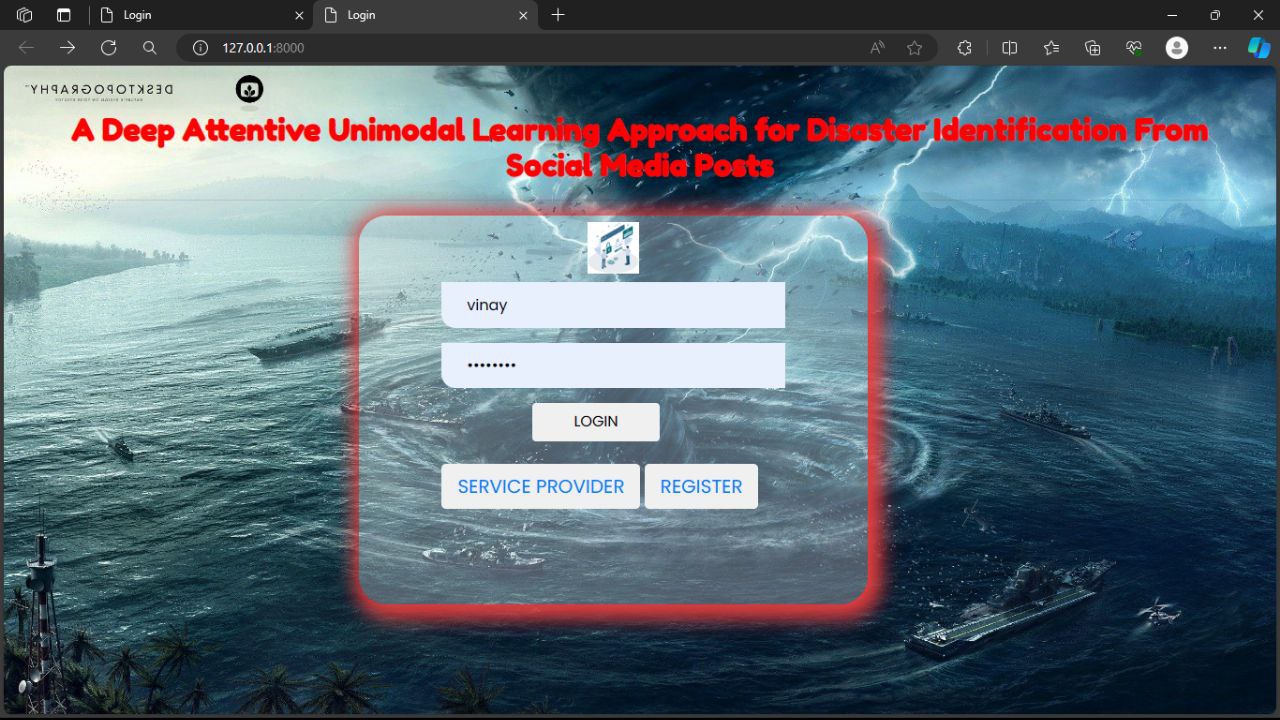
|  |  |  |  |
| --- | --- | --- | --- |
| **S.no** | **Test Case** | **Excepted Result** | **Result** |
| 1 | User register | User register successfully | Pass |
| 2 | User login | User login  successfully | Pass |
| 3 | Add Data | Add Data  Successfully | Pass |
| 4 | Data View | Data View Successfully | Pass |
| 5 | Data Descriptions | Data Descriptions Successfully | Pass |
| 6 | Correlation Matrix | Correlation Matrix Successfully | Pass |
| 7 | Box plot | Box plot Successfully | Pass |
| 8 | Heat Frequency graph | Successfully get  Heat Frequency graph | Pass |
| 9 | Distribution on Age graph | Successfully get  Distribution on Age graph | Pass |
| 10 | Calculated Frequency | Successfully get  Calculated Frequency | Pass |
| 11 | Scatter matrix | Successfully get  Scatter matrix | Pass |
| 12 | Age Wise Disease Scatter graph | Successfully get  Age Wise Disease Scatter graph | Pass |
| 13 | Cleaned data | Successfully get  Cleaned data | Pass |
| 14 | Random Forest Confusion matrix | Successfully get  Random Forest Confusion matrix | Pass |
| 15 | Decision Tree  Confusion matrix | Successfully get  Decision Tree  Confusion matrix | Pass |
| 16 | Random Forest Roc | Successfully get  Random Forest Roc | Pass |
| 17 | SVC Curve | Successfully get  Successfully get | Pass |
| 18 | Algorithm Accuracy | Algorithm Accuracy  Successfully | Pass |

**CHAPTER – 8**

**OUTPUT SCREENS**

**8.2 For service provider:**

In below screen click on ‘Service Provider’ link to get next screen



**Fig8.2.1: Login Pages Overview**

**Emphasize user authentication and access control.**

In below screen Service Provider has to login by using valid user name and password

After login will get next screen.

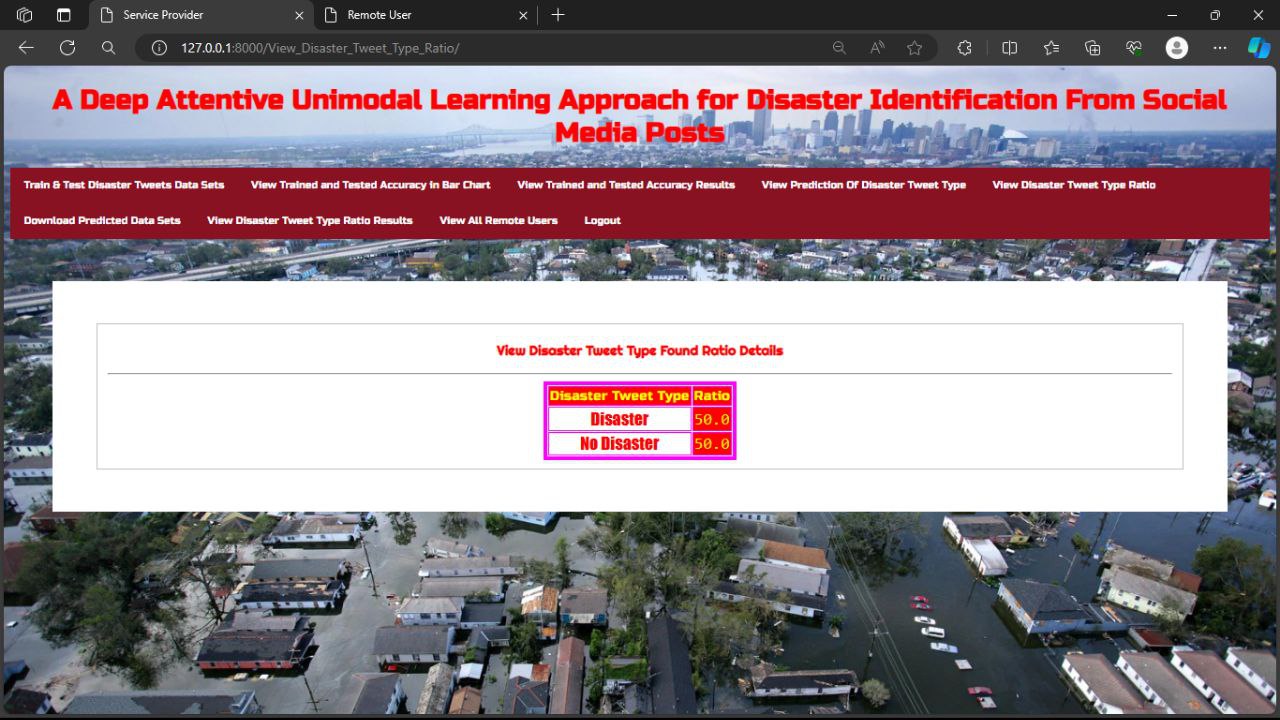


**Fig8.2.2: Showcase entry points for service providers**

In below screen after login the admin have the Access to datasets, Model accuracy visualization, Disaster tweet prediction, User management.

**Operations:**

* Train & Test Disaster Tweets Data Sets,
* View Trained and Tested Accuracy (Bar Chart),
* View Prediction Of Disaster Tweet Type,
* Download Predicted Data Sets,
* View Disaster Tweet Type Ratio,
* View All Remote Users.

**Fig 8.2.3: Main Menu**

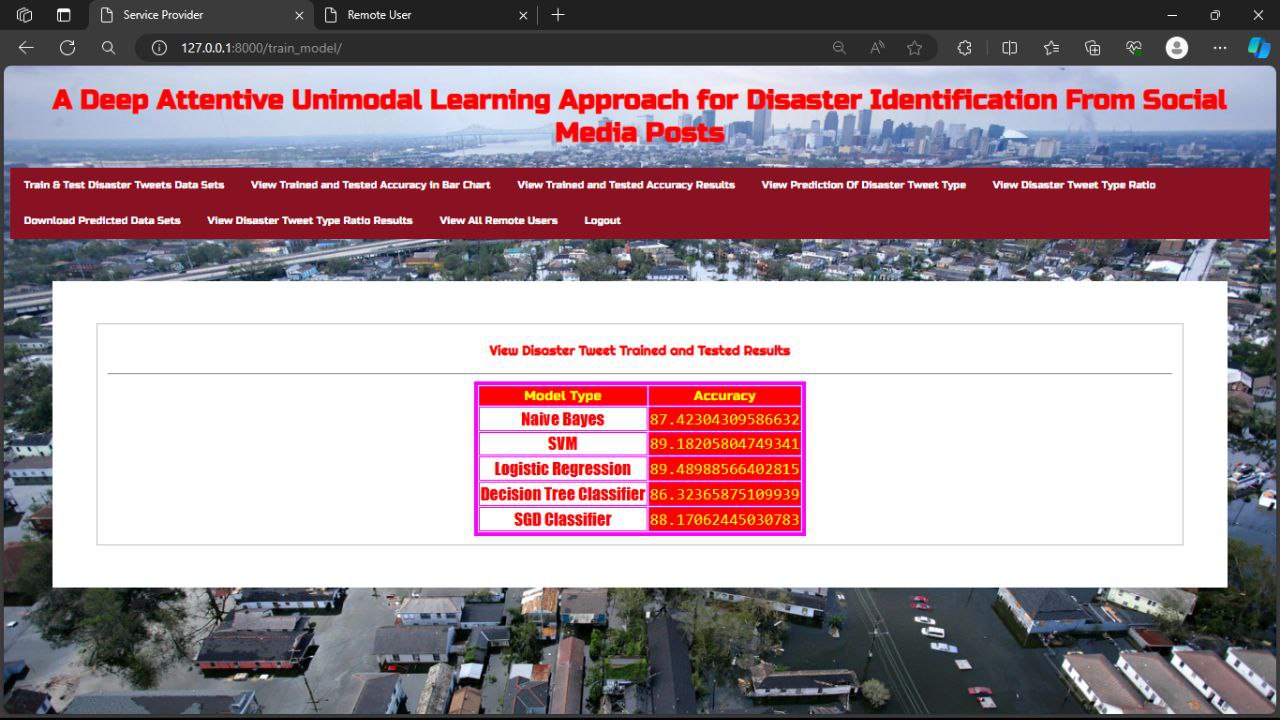
In below screen have the **Train and test disaster tweet data sets** details

**This Shows Model Accuracy Details:**

* Displays accuracy scores per model.
* Highlights consistency over iterations.
* Provides valuable performance insight.

In below screen click on ‘**View Trained and tested accuracy in bar chart’** to get below screen

Which is an accuracy visualization as shown in below screen



**Figure 8.2.4: Train and test disaster tweet data sets**

In above screen it shows the accuracy visualization by clicking on the ‘**Trained and tested accuracy in bar chart’**

**Accuracy Visualization:**

The below screen Shows the model performance.

* Uses bar chart.
* Includes **Naive Bayes, SVM, Logistic Regression, Decision Tree, SGD Classifier.**

In below screen click on ‘**View trained and tested accuracy results’** and will getbelow screen having the chart as shown in below screen.



**Figure 8.2.5: Accuracy Visualization in bar chart**

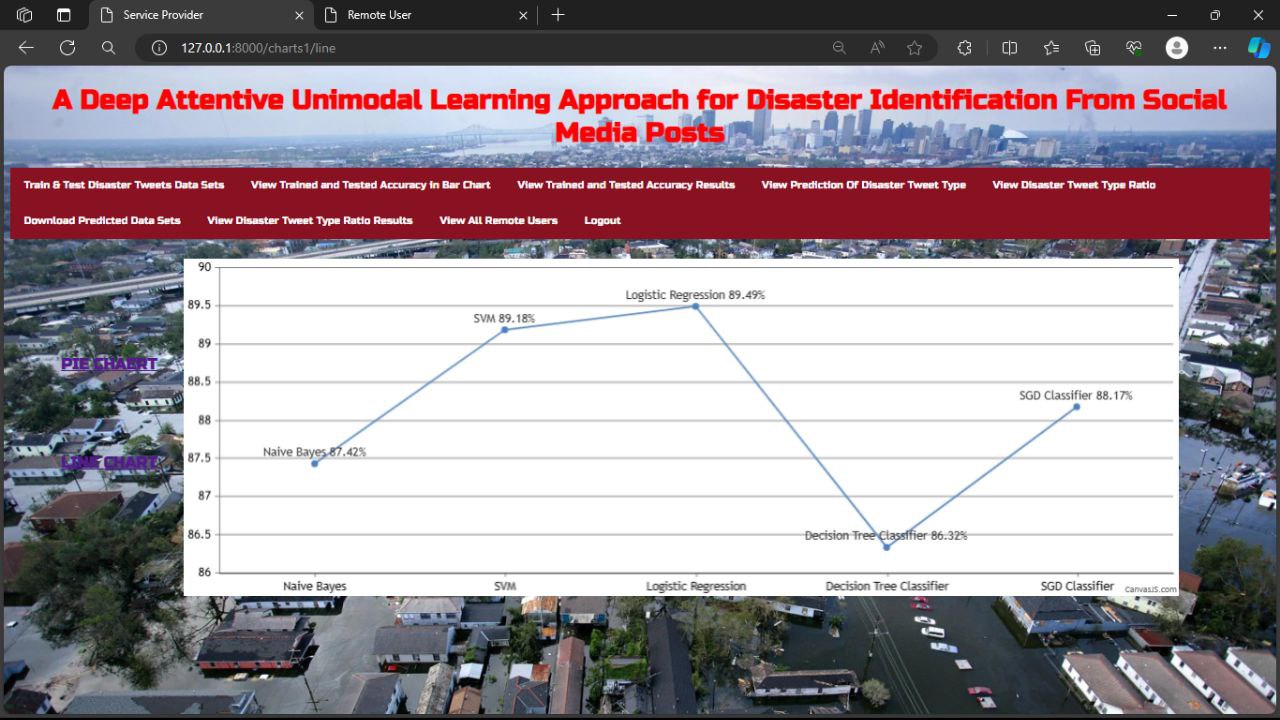
In below screen it shows the accuracy visualization by clicking on the ‘**View of trained and tested accuracy results’**

**Accuracy Visualization:**

It Shows the model performance.

* The above screen Uses line chart.
* Includes Naive Bayes, SVM, Logistic Regression, Decision Tree, SGD Classifier.

And then click on ‘pie chart’ in above screen button to get below screen



**Figure 8.2.6: View of trained and tested accuracy results in line chart.**

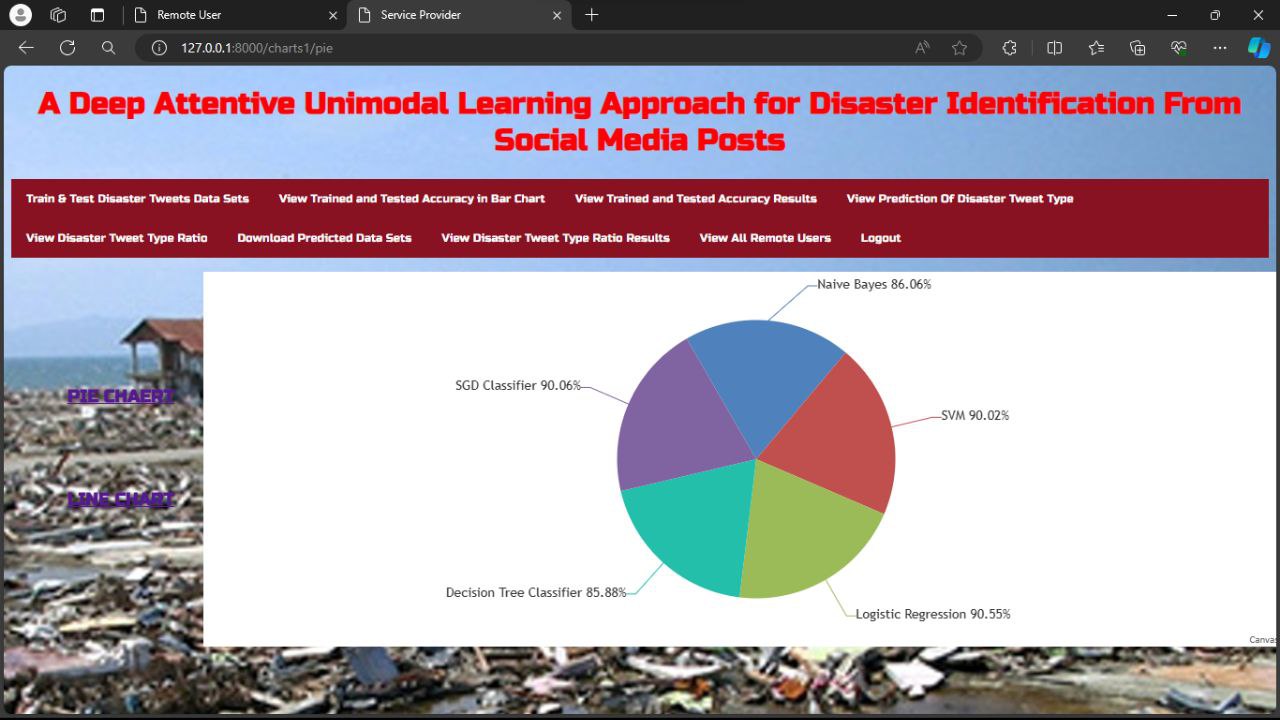
In above screen have the details of trained and tested accuracy in pie chart

**Accuracy Visualization:**

**It Shows the model performance.**

* Uses **Pie chart.**
* Includes **Naive Bayes, SVM, Logistic Regression, Decision Tree, SGD Classifier.**

In above screen click on ‘**View the Prediction Of Disaster Tweet Type’** button to get below screen fordetails.



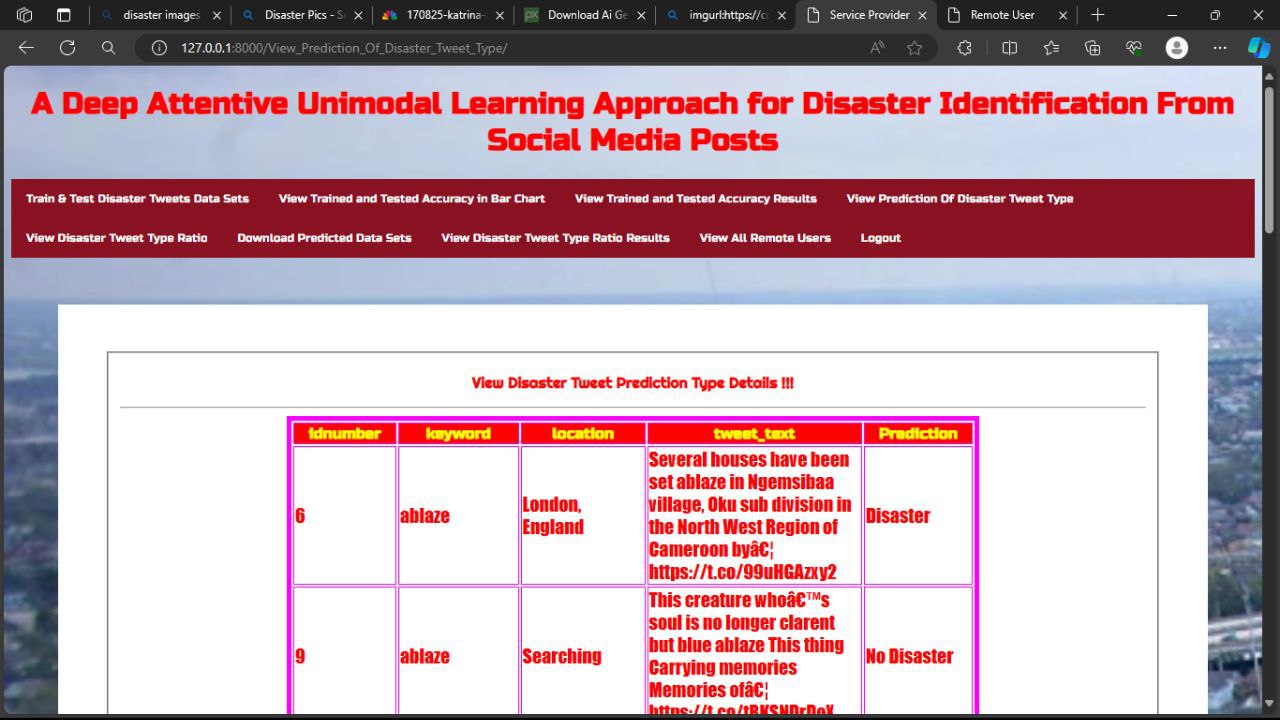
**Figure 8.2.7: View of trained and tested accuracy results in pie chart.**

In the above screen it shows tweeter post disaster prediction type

The above screen contains the details:

* **ID Number**
* **Keyword**
* **Location Tweet Text**
* **Prediction**

In above screen click on ‘**View Disaster Tweet Type Ratio’** button to get below screen fordetails



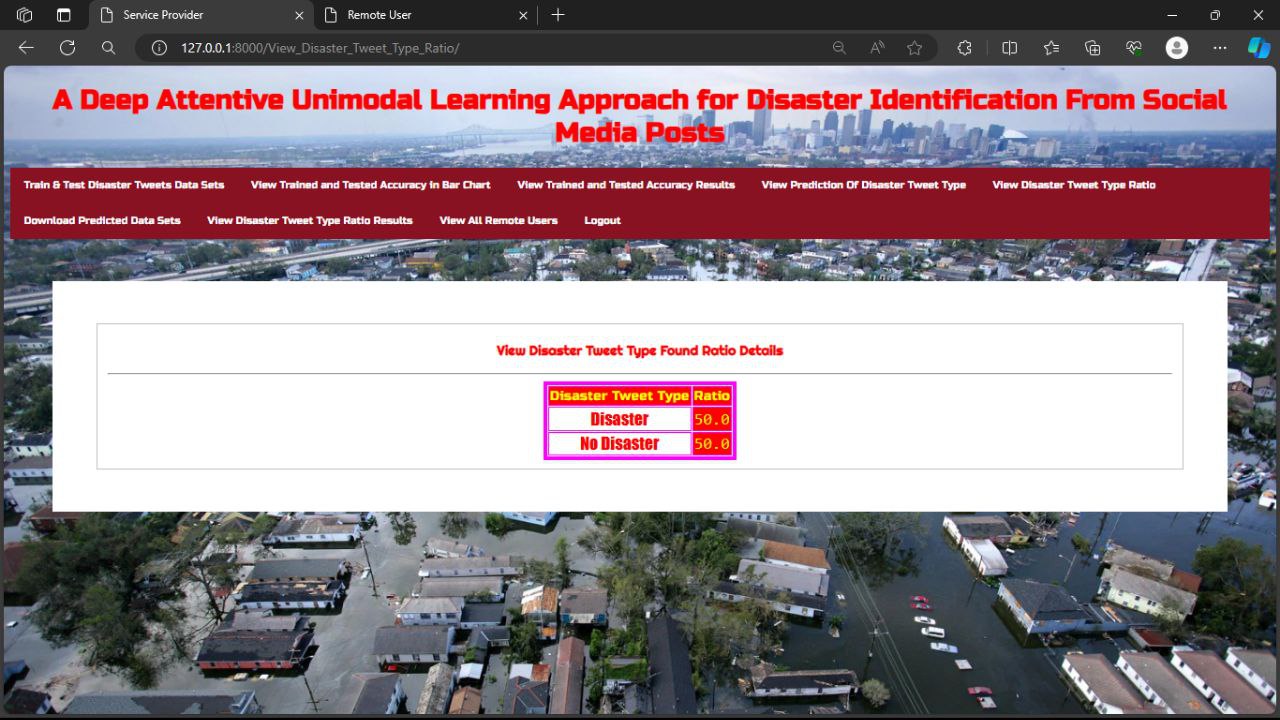
**Fig 8.2.8: View the Prediction of Disaster Tweet Type**

The above screen Displays distribution of disaster/non-disaster tweets.

And Offers insights into dataset balance.

It gives the disaster tweet type (DISASTER, NO DISASTER) ratio details

In above screen click on ‘**View Disaster Tweet Type Ratio Results’** button to get below screen in line chart.

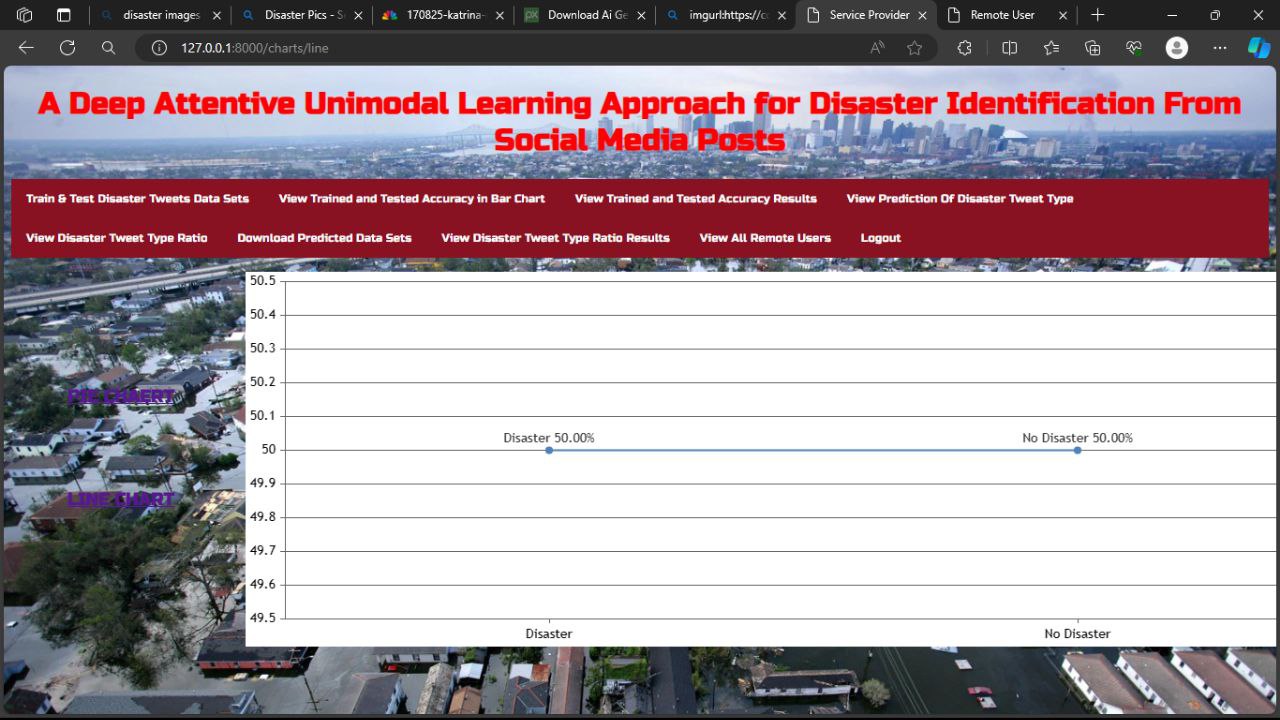


**Figure 8.2.9: View Disaster Tweet Type Ratio**

It gives the View of the Disaster Tweet Type (DISASTER, NO DISASTER) Ratio Results

in line chart

In above screen click on ‘**View All Remote Users’** button to get below screen fordetails

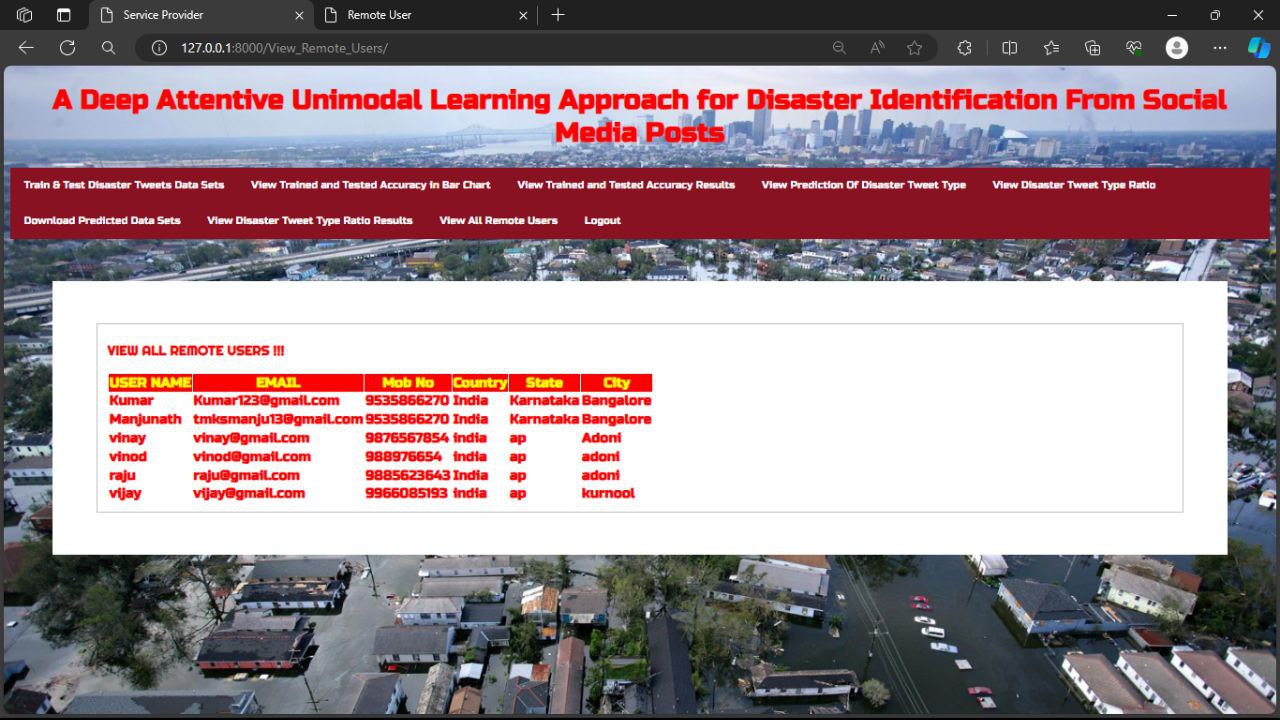


**Figure 8.2.10: View Disaster Tweet Type Ratio Results in line chart**

The above screen shows the **All Remote Users details** to service provider.

The screen contains the details of the users.

* User Name,
* email id,
* Mobile Number,
* Country,
* State,
* City



**Figure 8.2.11: View All Remote Users**

**Results & Evaluation Summary:**

* **High Accuracy:** Logistic Regression achieved >90% accuracy.
* **User-friendly Interface:** Easy navigation, data visualization, prediction.

**Future Directions:**

* Incorporate multimodal learning.
* Expand training data.
* Develop real-time monitoring capabilities.
* Integrate user feedback.

**Potential Impact:**

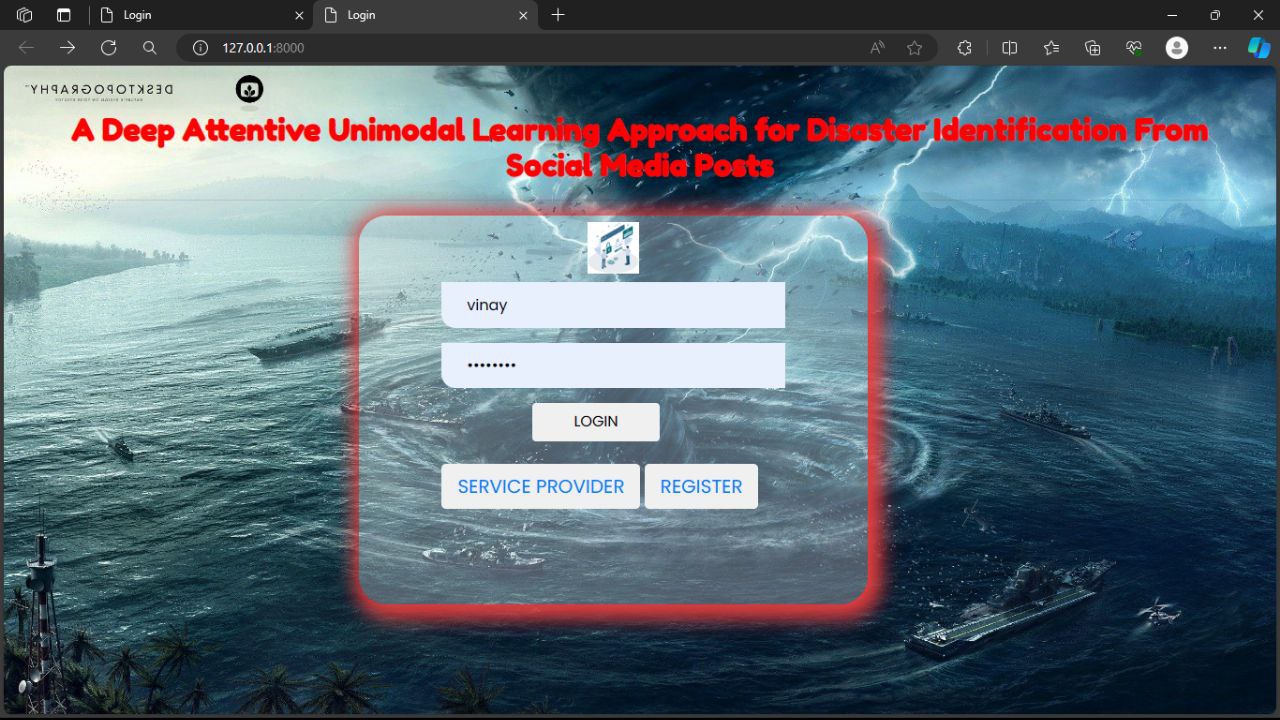
* Improve disaster response and management.
* Provide timely, accurate information.
* Contribute to saving lives and minimizing damage.

**8.1 Remote User:**

**Showcase entry points for service providers and users.**

**Emphasize user authentication and access control.**

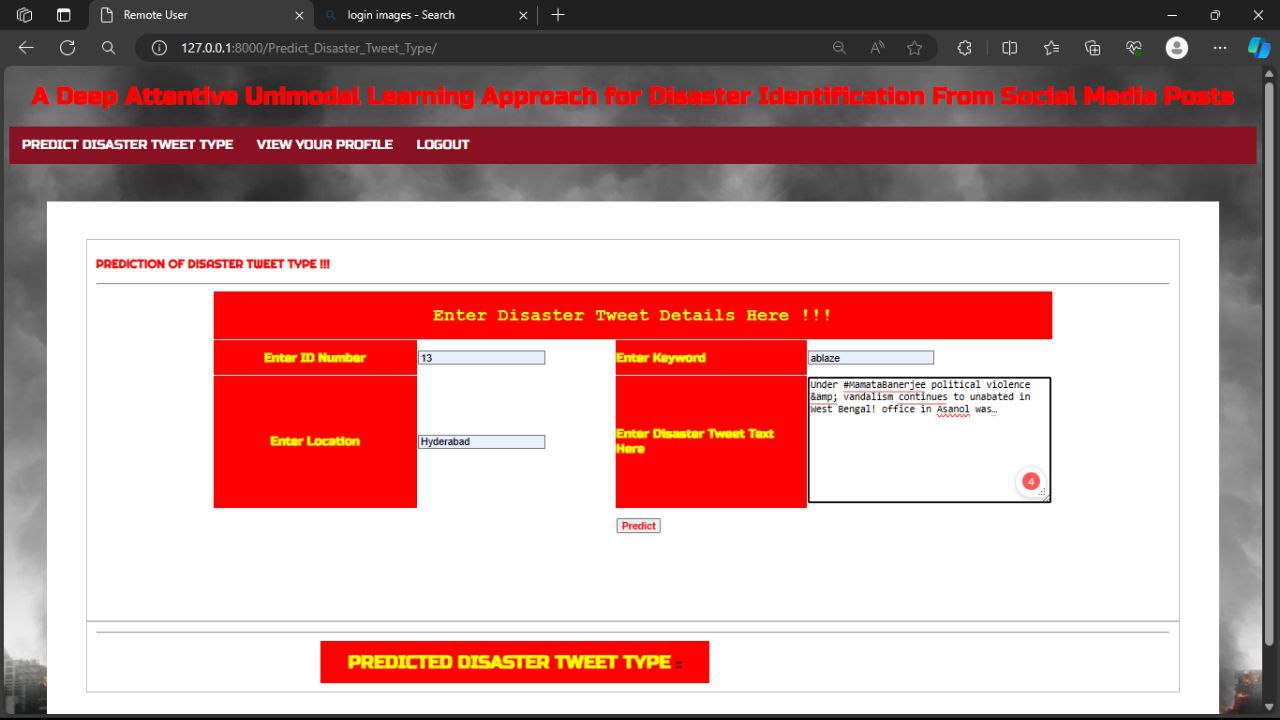
User registration required and then, in above screen Remote user has to login by using valid user name and password



**Fig8.1.1: Login Pages Overview**

After login will get below screen.

and then click on ‘**Predicted disaster tweet type** button to get below screen

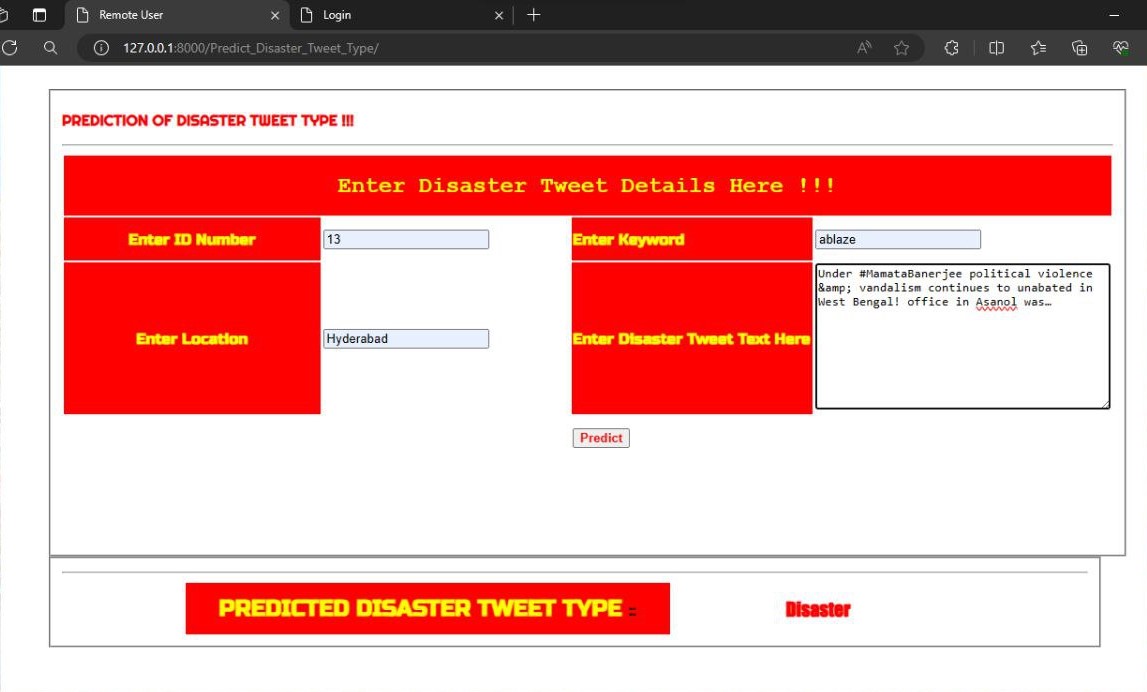
 **Figure 8.2.2: Prediction Interface:**

Shows system predicting disaster type.

**Inputs:** ID, keyword, location, text.

Highlights system's predictive capabilities.

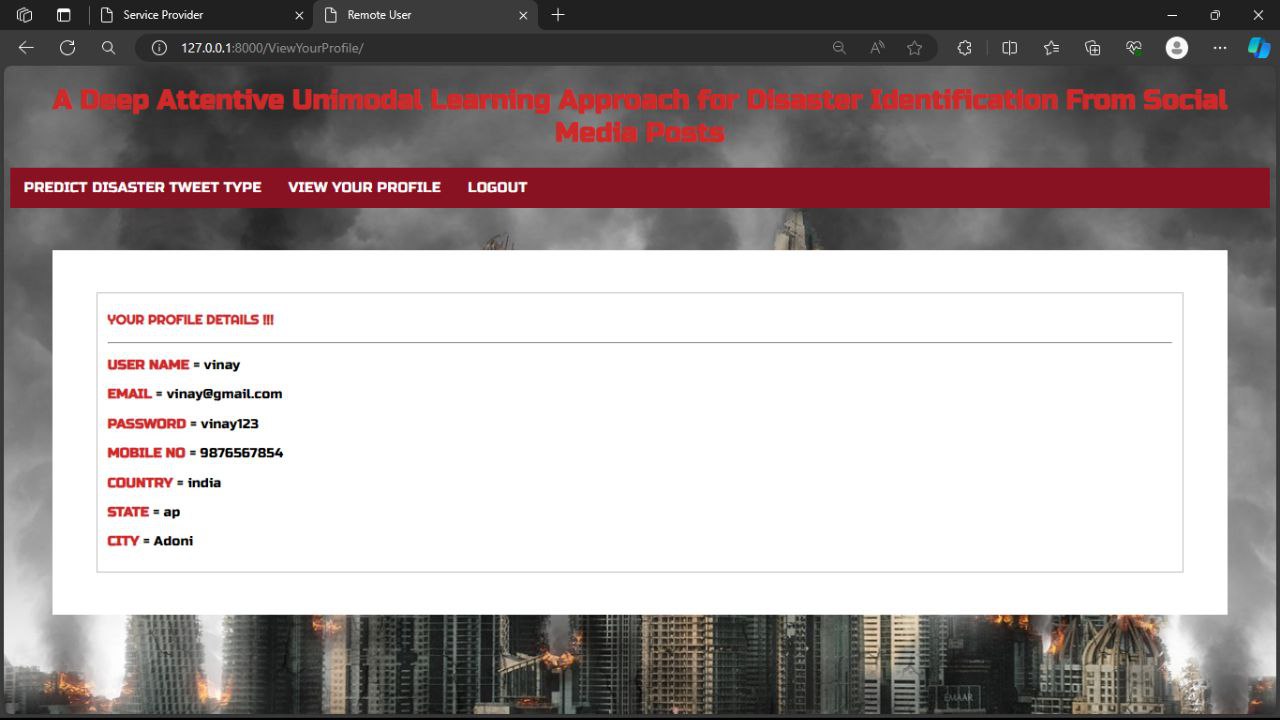
In the below screen the output is predicted for the given input details.



**Figure 8.2.3: Predicted disaster tweet type**

The profile contains the details of the user:

* **User Name,**
* **Email, Password,**
* **Mobile Number**
* **Country,**
* **State,**
* **City**



**Figure 8.2.4: Shows the profile of the user**

**CHAPTER – 9**

**CONCLUSION**

We have presented this project of unimodal approach that can effectively learn from the text data to classify the damage-related contents from Twitter. We utilize the pre-trained the attention mechanism with a BiLSTM model to extract the tweet features. The early fusion approach is used to aggregate both modalities’ features. Besides, this project investigated textual (i.e., BiLSTM, CNN, BiSTM+CNN, BiLSTM+ Attention) approaches for the baseline evaluation and constructed several unimodal models by exploiting them. The evaluation results revealed that our project proposed model outperforms the baseline unimodal models by acquiring the highest weighted F1-score of 93.21%. Moreover, the comparative analysis illustrated that the proposed method outcome is approximately 1% and 7% ahead of the existing start-of the- art models. Thus, the results confirmed the effectiveness of the proposed project method in identifying the disaster content based on unimodal information. The error analysis further showed that it is difficult to identify the damage and non-damage contents by analyzing only one modality. At the same time, intrinsic performance analysis elucidated that incorporating an attention mechanism boosts the overall performance.

**CHAPTER - 10**

**FUTURE ENHANCEMENT**

Despite achieving better performance unimodal approaches, there are still rooms for improving the proposed project method. In the future, we would like to explore different multimodal fusion approaches along with multitask learning technique for the disaster identification task. Besides, we aim to capture the combination of visual and textual features more effectively by employing the state of the art visual (i.e., Vision transformer [58]), textual (i.e., BERT [59], XLM-R [60]), and multimodal (i.e., VL- BERT [61], Visual BERT [62]) transformer models.

**CHAPTER -11**

**REFERENCES**

[1] K. K. Kapoor, K. Tamilmani, N. P. Rana, P. Patil, Y. K. Dwivedi, and S. Nerur, “Advances in social media research: Past, present and future,” Information Systems Frontiers, vol. 20, no. 3, pp. 531–558, 2018.

[2] J. Kim and M. Hastak, “Social network analysis: Characteristics of online social networks after a disaster,” International Journal of Information Management, vol. 38, no. 1, pp. 86–96, 2018.

[3] J. Son, H. K. Lee, S. Jin, and J. Lee, “Content features of tweets for effective communication during disasters: A media synchronicity theory perspective,” International Journal of Information Management, vol. 45, pp. 56–68, 2019.

[4] A. Elbanna, D. Bunker, L. Levine, and A. Sleigh, “Emergency management in the changing world of social media: Framing the research agenda with the stakeholders through engaged scholarship,” International Journal of Information Management, vol. 47, pp. 112–120, 2019.

[5] R. Dubey, A. Gunasekaran, S. J. Childe, D. Roubaud, S. F. Wamba, M. Giannakis, and C. Foropon, “Big data analytics and organizational culture as complements to swift trust and collaborative performance in the humanitarian supply chain,” International Journal of Production Economics, vol. 210, pp. 120–136, 2019.

[6] S. Akter and S. F. Wamba, “Big data and disaster management: a systematic review and agenda for future research,” Annals of Operations Research, vol. 283, no. 1, pp. 939–959, 2019.

[7] H. Mouzannar, Y. Rizk, and M. Awad, “Damage identification in social media posts using multimodal deep learning.” in ISCRAM, 2018.

[8] F. Ofli, F. Alam, and M. Imran, “Analysis of social media data using multimodal deep learning for disaster response,” arXiv preprint arXiv:2004.11838, 2020.

[9] F. Ofli, P. Meier, M. Imran, C. Castillo, D. Tuia, N. Rey, J. Briant, P. Millet, F. Reinhard, M. Parkan et al., “Combining human computing and machine learning to make sense of big (aerial) data for disaster response,” Big data, vol. 4, no. 1, pp. 47–59, 2016.

[10] P. Jain, B. Schoen-Phelan, and R. Ross, “Automatic flood detection in sentinei-2 images using deep convolutional neural networks,” in Proceedings of the 35th Annual ACM Symposium on Applied Computing, 2020, pp. 617–623.

[11] A. Kumar, J. P. Singh, Y. K. Dwivedi, and N. P. Rana, “A deep multimodal neural network for informative twitter content classification during emergencies,” Annals of Operations Research, pp. 1–32, 2020.

[12] T. Ghosh Mondal, M. R. Jahanshahi, R.-T. Wu, and Z. Y. Wu, “Deep learning-based multi-class damage detection for autonomous post-disaster reconnaissance,” Structural Control and Health Monitoring, vol. 27, no. 4, p. e2507, 2020.

[13] M. Imran, F. Ofli, D. Caragea, and A. Torralba, “Using ai and social media multimodal content for disaster response and management: Opportunities, challenges, and future directions,” 2020.

[14] L. Dwarakanath, A. Kamsin, R. A. Rasheed, A. Anandhan, and L. Shuib, “Automated machine learning approaches for emergency response and coordination via social media in the aftermath of a disaster: A review,” IEEE Access, vol. 9, pp. 68 917–68 931, 2021.

[15] C. Castillo, M. Imran, P. Meier, J. Lucas, J. Srivastava, H. Leson, F. Ofli,and P. Mitra, “Together we stand—supporting decision in crisis response: Artificial intelligence for digital response and micromappers,” by OCHA and partners. Istanbul: Tudor Rose, World Humanitarian Summit, pp. 93–95, 2016.

[16] C. Caragea, N. J. McNeese, A. R. Jaiswal, G. Traylor, H.-W. Kim, P. Mitra, D. Wu, A. H. Tapia, C. L. Giles, B. J. Jansen et al., “Classifying text messages for the haiti earthquake.” in ISCRAM. Citeseer, 2011.

[17] M. Imran, C. Castillo, J. Lucas, P. Meier, and S. Vieweg, “Aidr: Artificial intelligence for disaster response,” in Proceedings of the 23rd international conference on world wide web, 2014, pp. 159–162.

[18] Q. Huang and Y. Xiao, “Geographic situational awareness: mining tweets for disaster preparedness, emergency response, impact, and recovery,” ISPRS International Journal of Geo-Information, vol. 4, no. 3, pp. 1549– 1568, 2015.

[19] D. Nguyen, K. A. Al Mannai, S. Joty, H. Sajjad, M. Imran, and P. Mitra, “Robust classification of crisis-related data on social networks using convolutional neural networks,” in Proceedings of the International AAAI Conference on Web and Social Media, vol. 11, no. 1, 2017.

[20] O. Sharif, E. Hossain, and M. M. Hoque, “Combating hostility: Covid-19 fake news and hostile post detection in social media,” 2021.