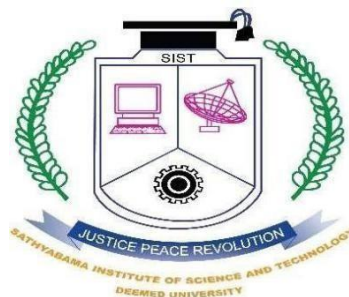


IMPROVISED DEPLOYMENT OF CATEGORY SEGMENTATION WITH KEYWORD FILTERING IN NEWS AND SOCIAL MEDIA USING BIG DATA

Submitted in Partial Fulfilment of The Requirements For
The Award Of
Bachelor of Engineering Degree in Computer Science and Engineering
By
ADABALA BALA VENKATESWARA RAO(Reg.no.37110011)
ATLURI KRISHNA TEJA(Reg.No.37110055)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade “A” by NAAC

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI –
600119**

MARCH 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NACC

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600119



www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this project report is the bonafide work of **ADABALA BALA VENKATESWARA RAO**(Reg No.37110011),**ATLURI KRISHNA TEJA**(Reg No.37110055) who carried out the project entitled "**IMPROVISED DEPLOYMENT OF CATEGORY SEGMENTATION WITH KEYWORD FILTERING IN NEWS AND SOCIAL MEDIA USING BIG DATA**" under my supervision from November 2020 to March 2021

Internal Guide

Dr. A.CHRISTY, MCA, Ph.D.,

Head of the Department

Dr. L. LAKSHMANAN M.E., Ph.D.,

Dr. S. VIGNESHWARI, M.E., Ph.D.,

Submitted for viva voce Examination held on_____.

Internal Examiner

External Examiner

DECLARATION

I, **ADABALA BALA VENKATESWARA RAO(Reg No.37110011)** hereby declare that the project report entitled **“IMPROVED DEPLOYMENT OF CATEGORY SEGMENTATION WITH KEYWORD FILTERING IN NEWS AND SOCIAL MEDIA USING BIG DATA”** done by me under the guidance of **Dr.A.CHRISTY MCA, Ph.D.**, is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Computer Science And Engineering.

DATE:

PLACE: CHENNAI.

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D., Dean**, School of Computing, **Dr. S. Vigneshwari M.E., Ph.D.** and **Dr. L. Lakshmanan M.E., Ph.D., Heads** of the Department of computer science and engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.A.CHRISTY, MCA, Ph.D.**, for his valuable guidance, suggestions and constant encouragement paved the way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

In the **Existing** system, When a significant event occurs, many news articles from different newsagents often report it. Moreover, these newsagents also provide platforms for their readers to write comments expressing their views or understanding. Through digesting these reader comments, we can gain insights into the reactions, suggestions, personal experiences, or public opinions with respect to the emerging event. In the **Proposed** system, unsupervised model called EFNS which is capable of capturing the intricate fine-grained associations among events, news, and comments. We also develop a multiplicative-update method to infer the parameters and prove the convergence of our algorithm. In the **Modification**, News are classified with respect to the category classified based on the set of keywords assigned to the centralized server. We use Machine Learning algorithm to principally classify the news according to the category.

TABLE OF CONTENTS

Chapter No.	Title	Page No.
	ABSTRACT	v
	LIST OF FIGURES	viii
1	INTRODUCTION	1
	1.1 Domain Introduction	1
	1.2 Project Introduction	10
	1.3 Objective of project	10
2	LITERATURE SURVEY	11
3	AIM AND SCOPE OF PRESENT INVESTIGATION	13
	3.1 Aim of the project	13
	3.2 Scope of the project	13
	3.3 EXISTING SYSTEM	13
	3.4 PROPOSED SYSTEM	
4	EXPERIMENTAL OR MATERIAL AND METHODS; ALOGRITHMS USED	14
	4.1 SYSTEM ARCITECTURE	14
	4.1.1 Architecture diagram	14
	4.1.2 Data flow diagram	14
	4.2 System Specification	16
	4.2.1Java Environment	16
	4.2.2 Hardware Environment	16
	4.2.3 Software Environment	16
	4.2.4 java	17

	4.2.5 Business Benefits	17
	4.2.6 Features of Java	17
	4.2.7 Java programming language	18
	4.2.8 Java platform	19
	4.2.9 Java virtual Machine	19
	4.2.10 Java API	19
	4.3 Back End	20
	4.3.1 What is mySQL	20
	4.4 Modules	24
	4.5 System testing	26
5	RESULT AND DISCUSSION	30
6	CONCLUSION AND FUTURE WORK	31
	6.1 CONCLUSION	31
	6.2 FUTURE WORK	31
	APPENDIX	32
	A. SOURCE CODE	32
	B. SCREEN SHOTS	43
	REFERENCES	49

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
1.1	CHARACTERISTICS OF BIG DATA	2
1.2	DATA TRANSFORMATION	6
4.1	ARCHITECTURE DIAGRAM	14
4.2	DATA FLOW DIAGRAM LEVEL0	14
4.3	DATA FLOW DIAGRAM LEVEL1	15
4.4	MYSQL ARCHITECTURE	21

CHAPTER – 1

INTRODUCTION

1.1 DOMAIN INTRODUCTION

Big data is an all-encompassing term for any collection of data sets so large and complex that it becomes difficult to process using traditional data processing applications. The challenges include analysis, capture, duration, search, sharing, storage, transfer, visualization, and privacy violations. The trend to larger data sets is due to the additional information derivable from analysis of a single large set of related data, as compared to separate smaller sets with the same total amount of data, allowing correlations to be found to "spot business trends, prevent diseases, combat crime and so on. So we can implement big data in our project because every employ has instructed information so we can make analysis on this data.

Characteristics Of Big Data

Big data is a term utilized to refer to the increase in the volume of data that are difficult to store, process, and analyze through traditional database technologies. The nature of big data is indistinct and involves considerable processes to identify and translate the data into new insights. The term "big data" is relatively new in IT and business. However, several researchers and practitioners have utilized the term in previous literature. For instance, referred to big data as a large volume of scientific data for visualization. Several definitions of big data currently exist." Meanwhile and defined big data as characterized by three Vs: volume, variety, and velocity. The terms volume, variety, and velocity were originally introduced by Gartner to describe the elements of big data challenges. IDC also defined big data technologies as "a new generation of technologies and architectures, designed to economically extract value from very large volumes of a wide variety of data, by enabling the high velocity capture, discovery, and/or analysis." specified that big data is not only characterized by the three Vs mentioned above but may also extend to four Vs, namely, volume, variety, velocity, and value This 4V definition is widely recognized because it highlights the meaning and necessity of big data.

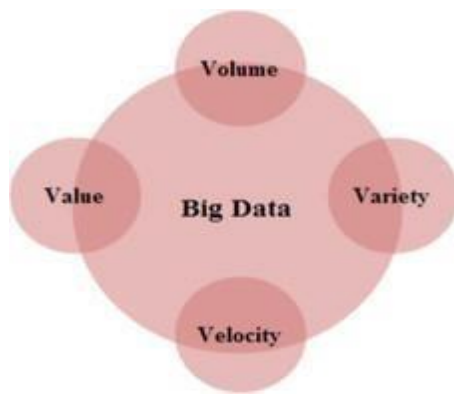


Fig 1.1 Characteristics Of Big Data

Volume - Refers to the amount of all types of data generated from different sources and continue to expand. The benefit of gathering large amounts of data includes the creation of hidden information and patterns through data analysis. Collecting longitudinal data requires considerable effort and underlying investments. Nevertheless, such mobile data challenge produced an interesting result similar to that in the examination of the predictability of human behavior patterns or means to share data based on human mobility and visualization techniques for complex data.

Variety-Refers to the different types of data collected via sensors, smart phones, or social networks. Such data types include video, image, text, audio, and data logs, in either structured or unstructured format. Most of the data generated from mobile applications are in unstructured format. For example, text messages, online games, blogs, and social media generate different types of unstructured data through mobile devices and sensors. Internet users also generate an extremely diverse set of structured and unstructured data.

Velocity -Refers to the speed of data transfer. The contents of data constantly change because of the absorption of complementary data collections, introduction of previously archived data or legacy collections, and streamed data arriving from multiple sources

Value- is the most important aspect of big data; it refers to the process of discovering huge hidden values from large datasets with various types and rapid generation

Classification Of Big Data

Big data are classified into different categories to better understand their characteristics shows the numerous categories of big data. The classification is important because of large-scale data in the cloud. The classification is based on five aspects: (i) data sources, (ii) content format, (iii) data stores, (iv) data staging, and (v) data processing.

Data sources include internet data, sensing and all stores of transnational information, ranges from unstructured to highly structured are stored in various formats. Most popular is the

relational database that come in a large number of varieties. As the result of the wide variety of data sources, the captured data differ in size with respect to redundancy, consistency and noise, etc.

Big Data Storage System

The rapid growth of data has restricted the capability of existing storage technologies to store and manage data. Over the past few years, traditional storage systems have been utilized to store data through structured RDBMS. However, almost storage systems have limitations and are inapplicable to the storage and management of big data. A storage architecture that can be accessed in a highly efficient manner while achieving availability and reliability is required to store and manage large datasets

Several storage technologies have been developed to meet the demands of massive data. Existing technologies can be classified as direct attached storage (DAS), network attached storage (NAS), and storage area network (SAN). In DAS, various hard disk drives (HDDs) are directly connected to the servers. Each HDD receives a certain amount of input/output (I/O) resource, which is managed by individual applications. Therefore, DAS is suitable only for servers that are interconnected on a small scale. Given the aforesaid low scalability, storage capacity is increased but expandability and upgradeability are limited significantly. NAS is a storage device that supports a network. NAS is connected directly to a network through a switch or hub via TCP/IP protocols. In NAS, data are transferred as files. Given that the NAS server can indirectly access a storage device through networks, the I/O burden on a NAS server is significantly lighter than that on a DAS server. NAS can orient networks, particularly scalable and bandwidth-intensive networks. Such networks include high-speed networks of optical-fiber connections. The SAN system of data storage is independent with respect to storage on the local area network (LAN). Multipath data switching is conducted among internal nodes to maximize data management and sharing. The organizational systems of data storages (DAS, NAS, and SAN) can be divided into three parts: (i) disc array, where the foundation of a storage system provides the fundamental guarantee, (ii) connection and network subsystems, which connect one or more disc arrays and servers, and (iii) storage management software, which oversees data sharing, storage management, and disaster recovery tasks for multiple servers.

Hadoop Background

Hadoop is an open-source Apache Software Foundation project written in Java that enables the distributed processing of large datasets across clusters of commodity. Hadoop has two primary components, namely, HDFS and Map Reduce programming framework. The most

significant feature of Hadoop is that HDFS and Map Reduce are closely related to each other; each are co-deployed such that a single cluster is produced. Therefore, the storage system is not physically separated from the processing system.

HDFS is a distributed file system designed to run on top of the local file systems of the cluster nodes and store extremely large files suitable for streaming data access. HDFS is highly fault tolerant and can scale up from a single server to thousands of machines, each offering local computation and storage. HDFS consists of two types of nodes, namely, a name node called “master” and several data nodes called “slaves.” HDFS can also include secondary name nodes. The name node manages the hierarchy of file systems and director namespace (i.e., metadata). File systems are presented in a form of name node that registers attributes, such as access time, modification, permission, and disk space quotas. The file content is split into large blocks, and each block of the file is independently replicated across data nodes for redundancy and to periodically send a report of all existing blocks to the name node.

Map Reduce is a simplified programming model for processing large numbers of datasets pioneered by Google for data-intensive applications. The Map Reduce model was developed based on GFS is adopted through open-source Hadoop implementation, which was popularized by Yahoo. Apart from the Map Reduce framework, several other current open-source Apache projects are related to the Hadoop ecosystem, including Hive, Hbase, Mahout, Pig, Zookeeper, Spark, and Avro. Twister provides support for efficient and iterative MapReduce computations. An overview of current Map Reduce projects and related software is shown in MapReduce allows an inexperienced programmer to develop parallel programs and create a program capable of using computers in a cloud. In most cases, programmers are required to specify two functions only: the map function (mapper) and the reduce function (reducer) commonly utilized in functional programming. The mapper regards the key/value pair as input and generates intermediate key/value pairs. The reducer merges all the pairs associated with the same (intermediate) key and then generates an output. Summarizes the process of the map/reduce function.

Map Reduce In Clouds

Map Reduce accelerates the processing of large amounts of data in a cloud; thus, Map Reduce, is the preferred computation model of cloud providers. Map Reduce is a popular cloud computing framework that robotically performs scalable distributed applications and provides an interface that allows for parallelization and distributed computing in a cluster of servers. The approach is to apply scientific computing problems to the MapReduce framework where scientists can efficiently utilize existing resources in the cloud to solve computationally large-scale scientific data.

Currently, many alternative solutions are available to deploy MapReduce in cloud environments; these solutions include using cloud MapReduce runtimes that maximize cloud infrastructure services, using MapReduce as a service, or setting up one's own MapReduce cluster in cloud instances. Several strategies have been proposed to improve the performance of big data processing. Moreover, effort has been exerted to develop SQL interfaces in the MapReduce framework to assist programmers who prefer to use SQL as a high-level language to express their task while leaving all of the execution optimization details to the backend

Research Challenges

Although cloud computing has been broadly accepted by many organizations, research on big data in the cloud remains in its early stages. Several existing issues have not been fully addressed. Moreover, new challenges continue to emerge from applications by organization. In the subsequent sections, some of the key research challenges, such as scalability, availability, data integrity, data transformation, data quality, data heterogeneity, privacy and legal issues, and regulatory governance, are discussed.

Scalability

Scalability is the ability of the storage to handle increasing amounts of data in an appropriate manner. Scalable distributed data storage systems have been a critical part of cloud computing infrastructures. The lack of cloud computing features to support RDBMSs associated with enterprise solutions has made RDBMSs less attractive for the deployment of large-scale applications in the cloud. This drawback has resulted in the popularity of NoSQL

A NoSQL database provides the mechanism to store and retrieve large volumes of distributed data. The features of NoSQL databases include schema-free, easy replication support, simple API, and consistent and flexible modes. Different types of NoSQL databases, such as key-value column-oriented, and document-oriented, provide support for big data. Shows a comparison of various NoSQL database technologies that provide support for large datasets.

Availability

Availability refers to the resources of the system accessible on demand by an authorized individual. In a cloud environment, one of the main issues concerning cloud service providers is the availability of the data stored in the cloud. For example, one of the pressing demands on cloud service providers is to effectively serve the needs of the mobile user who requires single or multiple data within a short amount of time. Therefore, services must remain operational even in the case of a security breach. In addition, with the increasing number of cloud users, cloud service

providers must address the issue of making the requested data available to users to deliver high-quality services. Lee et al. introduced a multi-cloud model called “rain clouds” to support big data exploitation. “Rain clouds” involves cooperation among single clouds to provide accessible resources in an emergency. Schroeck et al. predicted that the demand for more real time access to data may continue to increase as business models evolve and organizations invest in technologies required for streaming data and smart phones.

Data Integrity

A key aspect of big data security is integrity. Integrity means that data can be modified only by authorized parties or the data owner to prevent misuse. The proliferation of cloud-based applications provides users the opportunity to store and manage their data in cloud data centers. Such applications must ensure data integrity. However, one of the main challenges that must be addressed is to ensure the correctness of user data in the cloud.

Transformation

Transforming data into a form suitable for analysis is an obstacle in the adoption of big data. Owing to the variety of data formats, big data can be transformed into an analysis workflow in two ways

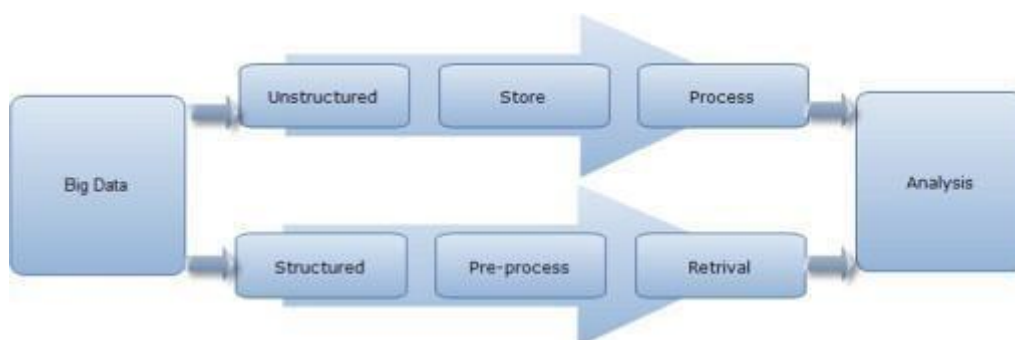


Fig 1.2 Data Transformation

Data Quality

In the past, data processing was typically performed on clean datasets from well-known and limited sources. Therefore, the results were accurate. However, with the emergence of big data, data originate from many different sources; not all of these sources are well-known or verifiable. Poor data quality has become a serious problem for many cloud service providers because data are often collected from different sources. For example, huge amounts of data are generated from smart phones, where inconsistent data formats can be produced as a result of heterogeneous sources. The data quality problem is usually defined as “any difficulty encountered

along one or more quality dimensions that render data completely or largely unfit for use". Therefore, obtaining high-quality data from vast collections of data sources is a challenge. High-quality data in the cloud is characterized by data consistency. If data from new sources are consistent with data from other sources, then the new data are of high quality.

Heterogeneity

Variety, one of the major aspects of big data characterization, is the result of the growth of virtually unlimited different sources of data. This growth leads to the heterogeneous nature of big data. Data from multiple sources are generally of different types and representation forms and significantly interconnected; they have incompatible formats and are inconsistently represented.

In a cloud environment, users can store data in structured, semi-structured, or unstructured format. Structured data formats are appropriate for today's database systems, whereas semi-structured data formats are appropriate only to some extent. Unstructured data are inappropriate because they have a complex format that is difficult to represent in rows and columns. According to Kocarev and Jakimoski, the challenge is how to handle multiple data sources and types.

Privacy

Privacy concerns continue to hamper users who outsource their private data into the cloud storage. This concern has become serious with the development of big data mining and analytics, which require personal information to produce relevant results, such as personalized and location-based services. Information on individuals is exposed to scrutiny, a condition that gives rise to concerns on profiling, stealing, and loss of control.

Legal/Regulatory Issues

Specific laws and regulations must be established to preserve the personal and sensitive information of users. Different countries have different laws and regulations to achieve data privacy and protection. In several countries, monitoring of company staff communications is not allowed. However, electronic monitoring is permitted under special circumstances. Therefore, the question is whether such laws and regulations offer adequate protection for individuals' data while enjoying the many benefits of big data in the society at large.

Governance

Data governance embodies the exercise of control and authority over data-related rules of law, transparency, and accountabilities of individuals and information systems to achieve business

objectives. The key issues of big data in cloud governance pertain to applications that consume massive amounts of data streamed from external sources. Therefore, a clear and acceptable data policy with regard to the type of data that need to be stored, how quickly an individual needs to access the data, and how to access the data must be defined.

Big data governance involves leveraging information by aligning the objectives of multiple functions, such as telecommunication carriers having access to vast troves of customer information in the form of call detail records and marketing seeking to monetize this information by selling it to third parties.

Moreover, big data provides significant opportunities to service providers by making information more valuable. However, policies, principles, and frameworks that strike stability between risk and value in the face of increasing data size and deliver better and faster data management technology can create huge challenges.

Cloud governance recommends the use of various policies together with different models of constraints that limit access to underlying resources. Therefore, adopting governance practices that maintain a balance between risk exposure and value creation is a new organizational imperative to unlock competitive advantages and maximize value from the application of big data in the cloud.

Open Research Issues

Numerous studies have addressed a number of significant problems and issues pertaining to the storage and processing of big data in clouds. The amount of data continues to increase at an exponential rate, but the improvement in the processing mechanisms is relatively slow. Only a few tools are available to address the issues of big data processing in cloud environments. State-of-the-art techniques and technologies in many important big data applications (i.e., MapReduce, Dryad, Pregel, PigLatin, MangoDB, Hbase, SimpleDB, and Cassandra) cannot solve the actual problems of storing and querying big data. For example, Hadoop and Map Reduce lack query processing strategies and have low-level infrastructures with respect to data processing and management. Despite the plethora of work performed to address the problem of storing and processing big data in cloud computing environments, certain important aspects of storing and processing big data in cloud computing are yet to be solved. Some of these issues are discussed in the subsequent subsections.

Data Staging

The most important open research issue regarding data staging is related to the

heterogeneous nature of data. Data gathered from different sources do not have a structured format. For instance, mobile cloud-based applications, blogs, and social networking are inadequately structured similar to pieces of text messages, videos, and images. Transforming and cleaning such unstructured data before loading those into the warehouse for analysis are challenging tasks. Efforts have been exerted to simplify the transformation process by adopting technologies such as Hadoop and MapReduce to support the distributed processing of unstructured data formats. However, understanding the context of unstructured data is necessary, particularly when meaningful information is required. MapReduce programming model is the most common model that operates in clusters of computers; it has been utilized to process and distribute large amounts of data.

Distributed Storage Systems

Numerous solutions have been proposed to store and retrieve massive amounts of data. Some of these solutions have been applied in a cloud computing environment. However, several issues hinder the successful implementation of such solutions, including the capability of current cloud technologies to provide necessary capacity and high performance to address massive amounts of data, optimization of existing file systems for the volumes demanded by data mining applications, and how data can be stored in such a manner that they can be easily retrieved and migrated between servers.

Data Analysis

The selection of an appropriate model for large-scale data analysis is critical. Talia pointed out that obtaining useful information from large amounts of data requires scalable analysis algorithms to produce timely results. However, current algorithms are inefficient in terms of big data analysis. Therefore, efficient data analysis tools and technologies are required to process such data. Each algorithm performance ceases to increase linearly with increasing computational resources. As researchers continue to probe the issues of big data in cloud computing, new problems in big data processing arise from the transitional data analysis techniques. The speed of stream data arriving from different data sources must be processed and compared with historical information within a certain period of time. Such data sources may contain different formats, which makes the integration of multiple sources for analysis a complex task .

Data Security

Although cloud computing has transformed modern ICT technology, several unresolved security threats exist in cloud computing. These security threats are magnified by the volume,

velocity, and variety of big data. Moreover, several threats and issues, such as privacy, confidentiality, integrity, and availability of data, exist in big data using cloud computing platforms. Therefore, data security must be measured once data are outsourced to cloud service providers. The cloud must also be assessed at regular intervals to protect it against threats. Cloud vendors must ensure that all service level agreements are met. Recently, some controversies have revealed how some security agencies use data generated by individuals for their own benefit without permission.

Therefore, policies that cover all user privacy concerns should be developed. Traditionally, the most common technique for privacy and data control is to protect the systems utilized to manage data rather than the data itself; however, such systems have proven to be vulnerable. Utilizing strong cryptography to encapsulate sensitive data in a cloud computing environment and developing a novel algorithm that efficiently allows for key management and secure key exchange are important to manage access to big data, particularly as they exist in the cloud independent of any platform. Moreover, the issue with integrity is that previously developed hashing schemes are no longer applicable to large amounts of data. Integrity verification is also difficult because of the lack of support, given remote data access and the lack of information on internal storage.

1.2 PROJECT INTRODUCTION:

Using a language statistical/mathematical model to automatically classify news articles is useful for automatization of subscription services, recommender systems, as well as dynamic content filtering and ordering. In this paper we study the case of news article classification, but the models and processes can be applied to other contents, such as regulations, education, and jurisprudence. Machine learning techniques has been applied for sentence/text/document classification in the context of binary classification (email is spam or "ham" [1]), sentiment analysis (reviews are positive or negative [2]) or multi-label classification, which is our focus of study. Although many studies are available for English documents or articles, other languages, such as Chinese, still do not have many references. Thus, we also study the differences in the text analysis related to Chinese language and the additional care that is needed when dealing with Chinese characters.

1.3 OBJECTIVE OF THE PROJECT:

The objective of the project is to classify the news based on user interest and likes.

CHAPTER – 2

LITERATURE SURVEY

Paper 1: Tagging your tweets: A probabilistic modeling of hashtag annotation in twitter

Author:

Tom Rosenstiel, Jeff Sonderman, Kevin Loker, Maria Ivancin And Nina Kjarval

Abstract:

Twitter users also tend to be younger than social media users in general. They tend to use the service heavily—the majority several times a day—and their use of the network is increasing how much news they consume. Almost all Twitter users are also consumers of other forms of news media. While Twitter users follow news in general on the service, and sometimes do so just as a way of passing time, they act differently when they are following breaking news, becoming even more participatory—commenting, posting and sharing at moments when events are moving fastest. All of these offer signals for how news publishers can make more effective use of social networks in general and Twitter in particular. These are some of the findings of a comprehensive survey of Twitter users as well as social media users who are not on Twitter, produced by the American Press Institute and Twitter in collaboration with research firm DB5. The study, conducted online, surveyed 4,713 people — 3,713 Twitter users and, for comparison, a separate sample of 1,000 nationally representative social media users, including 469 who were not on Twitter. The findings come at time when Twitter is increasing its emphasis on news and making tweaks that may affect publishers' strategies. At the end of this report, API has included a set of recommendations for publishers based on the survey findings.

Paper 2: Hashtag Recommendation Using Dirichlet Process Mixture Models Incorporating Types of Hashtags

Author:

Yeyun Gong, Qi Zhang, Xuanjing Huang

Abstract:

In recent years, the task of recommending hashtags for microblogs has been given increasing attention. Various methods have been proposed to study the problem from different aspects. However, most of the recent studies have not considered the differences in the types or uses of hashtags. In this paper, we introduce a novel nonparametric Bayesian method for this task. Based on the Dirichlet Process Mixture Models (DPMM), we incorporate the type of hashtag

as a hidden variable. The results of experiments on the data collected from a real world microblogging service demonstrate that the proposed method outperforms state-of-the-art methods that do not consider these aspects. By taking these aspects into consideration, the relative improvement of the proposed method over the state-of-the-art methods is around 12.2% in F1-score.

Paper 3: Bridging Vocabularies to Link Tweets and News

Author:

Tuan-Anh Hoang-Vu, Aline Bessa

Abstract:

Social media has become a popular platform for publishing, sharing and consuming news. However, it is not a replacement for traditional sources of news | they are complementary. While news sites provide in-depth and comprehensive coverage of events and topics, social media postings include comments, opinions and rumors about facts publicized on the news. Social media can thus serve as a useful sensor for how popular a story (or topic) is, for how long, and people's sentiments about it. To use social media as a sensor, we first need to associate postings to news stories and topics. But doing so is challenging since postings are short and the vocabularies used in postings and in news can be very different. In this paper, we take a first step towards addressing this problem. We propose a framework that uses news as a proxy to build a topic model and associates Twitter postings (a.k.a. tweets) to the derived topics. Subsequently, to deal with vocabulary differences, we present a new strategy to adapt the topic model derived from news to tweets. We report the results of an experimental evaluation which indicates that our framework obtains high accuracy for a variety of topics, and that domain adaptation leads to significant gains in both precision and recall.

Paper 4: Text Understanding from Scratch

Author:

Xiang Zhang, Yann LeCun

Abstract:

This article demonstrates that we can apply deep learning to text understanding from character-level inputs all the way up to abstract text concepts, using temporal convolutional networks (LeCun et al., 1998) (ConvNets). We apply ConvNets to various large-scale datasets, including ontology classification, sentiment analysis, and text categorization. We show that temporal ConvNets can achieve astonishing performance without the knowledge of words, phrases, sentences and any other syntactic or semantic structures with regards to a human language. Evidence shows that our models can work for both English and Chinese.

CHAPTER – 3

AIM AND SCOPE OF PRESENT INVESTIGATION

3.1 AIM OF THE PROJECT

The aim of the project is to find out the news very easily using keywords classification.

3.2 SCOPE OF THE PROJECT

The news will be categorised using keywords abstracted from them and the user will be able to view the news he likes

3.3 EXISTING SYSTEM

There is no Automatic process to segregate the news and show to the users. All the news are displayed to all the user without any segregation, so there are more changes for the users to miss out some important News even.

DISADVANTAGES:

- Delay of news updation
- There is no segregation of news

3.4 PROPOSED SYSTEM

News are classified with respect to the category classified based on the set of keywords assigned to the centralized server. We use Machine Learning algorithm to principally classify the news according to the category. We assign the keywords like Ball, Bat, Dhoni, ViratKohli, Ground and other related words to the Sports Category, same way rest of the keywords are assigned to the corresponding categories. We assign synonyms into this category which is very will be effective in the proper segregation of the messages. We also add Automatic Alert system in this Application about a particular Keyword or its synonyms so that automatic notification is intimated to the customer even he / she misses out to view the News.

ADVANTAGES:

- Quick updation of news
- User interest based news updation
- Top listed user likes news.

CHAPTER – 4

EXPERIMENTAL OR MATERIAL METHODS; ALGORITHMS USED

4.1 SYSTEM ARCHITECTURE

4.1.1 Architecture Diagram

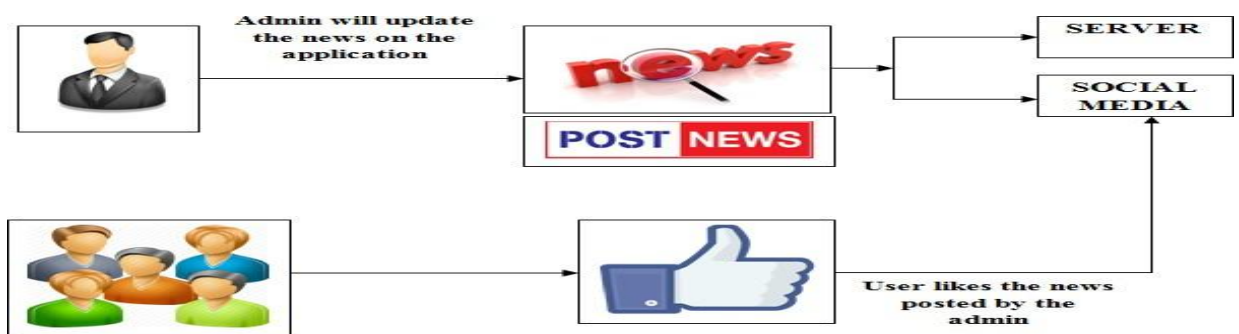


Fig 4.1 Architecture Diagram

4.1.2 DATA FLOW DIAGRAM:

LEVEL 0:

A data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various subprocesses the data moves through. DFDs are built using standardized symbols and notation to describe various entities and their relationships.

Data flow diagrams visually represent systems and processes that would be hard to describe in a chunk of text. You can use these diagrams to map out an existing system and make it better or to plan out a new system for implementation. Visualizing each element makes it easy to identify inefficiencies and produce the best possible system.

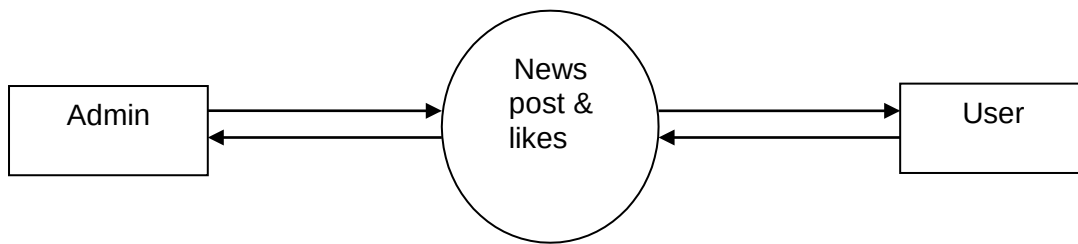


Fig 4.1.2 Data flow diagram level 0

LEVEL 1:

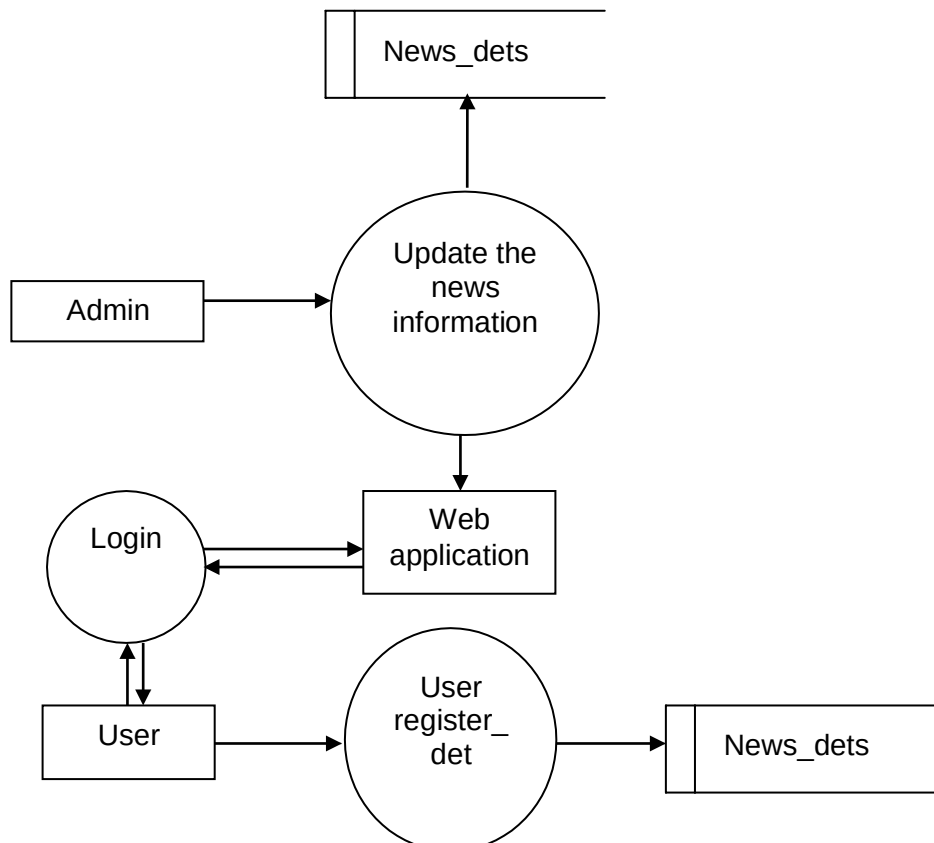


Fig 4.3 Data flow diagram level 1

4.2 SYSTEM SPECIFICATION

4.2.1 JAVA PROGRAMMING LANGUAGE:

REQUIREMENT ANALYSIS

Requirement analysis determines the requirements of a new system. This project analyses on product and resource requirement, which is required for this successful system. The product requirement includes input and output requirements it gives the wants in term of input to produce the required output. The resource requirements give in brief about the software and hardware that are needed to achieve the required functionality.

4.2.2 Hardware Environment

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It shows what the systems do and not how it should be implemented.

- Hard disk : 500 GB
- RAM : 4 GB
- PROCESSOR : CORE i5/ i7

4.2.3 Software Environment

The software requirements are the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the team's and tracking the team's progress throughout the development activity.

- Operating system : Windows 7/8.1
- Languages : Java , Solidity, go
- Data Base : Mysql , JSON
- IDE : Net Beans 8.2

4.2.4 Java

The Java platform is the ideal platform for network computing. Running across all platforms -- from servers to cell phones to smart cards -- Java technology unifies business infrastructure to create a seamless, secure, networked platform for your business. The Java platform benefits from a massive community of developers and supporters that actively work on delivering Java technology-based products and services as well as evolving the platform through an open, community-based, standards organization known as the Java Community Process program. You can find Java technology in cell phones, on laptop computers, on the Web, and even trackside at Formula One Grand Prix races. The fact is today, you can find Java technology just about everywhere!

4.2.5 Business Benefits

- A richer user experience - Whether you're using a Java technology-enabled mobile phone to play a game or to access your company's network, the Java platform provides the foundation for true mobility. The unique blend of mobility and security in Java technology makes it the ideal development and deployment vehicle for mobile and wireless solutions.
- The ideal execution environment for Web services - The Java and XML languages are the two most extensible and widely accepted computing languages on the planet, providing maximum reach to everyone, everywhere, every time, to every device and platform.
- Enabling business from end to end - Java offers a single, unifying programming model that can connect all elements of a business infrastructure.

4.2.6 FEATURES OF JAVA

Java is a programming language originally developed by James Gosling at Sun Microsystems (which is now a subsidiary of Oracle Corporation) and released in 1995 as a core component of Sun Microsystems' Java platform. The language derives much of its syntax from C and C++ but has a simpler object model and fewer low-level facilities. Java applications are typically compiled to byte code(class file) that can run on any Java Virtual Machine (JVM)

regardless of computer architecture. Java is general-purpose, concurrent, class-based, and object-oriented, and is specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere". Java is considered by many as one of the most influential programming languages of the 20th century, and widely used from application software to web application.

The original and reference implementation Java compilers, virtual machines, and class libraries were developed by Sun from 1995. As of May 2007, in compliance with the specifications of the Java Community Process, Sun relicensed most of their Java technologies under the GNU General Public License. Others have also developed alternative implementations of these Sun technologies, such as the GNU Compiler for Java and GNU Class path.

4.2.7 JAVA PROGRAMMING LANGUAGE

The Java programming language is a high-level language that can be characterized by all of the following buzzwords:

- Simple
- Architecture neutral
- Object oriented
- Portal
- Distributed
- High performance

Each of the preceding buzzwords is explained in *The Java Language Environment*, a white paper written by James Gosling and Henry Chilton. In the Java programming language, all source code is first written in plain text files ending with the .java extension. Those source files are then compiled into .class files by the javac compiler. A .class file does not contain code that is native to your processor; it instead contains *byte codes* — the machine language of the Java Virtual Machine (Java VM). The java launcher tool then runs your application with an instance of the Java Virtual Machine.

.

Because the Java VM is available on many different operating systems, the same .class files are capable of running on Microsoft Windows, the Solaris Operating System (Solaris OS), Linux, or Mac OS. Some virtual machines, such as the Java Hotspot virtual machine, perform additional steps at runtime to give your application a performance boost. This includes various

tasks such as finding performance bottlenecks and recompiling (to native code) frequently used sections of code.

4.2.8 JAVA PLATFORM

A *platform* is the hardware or software environment in which a program runs. We've already mentioned some of the most popular platforms like Microsoft Windows, Linux, Solaris OS, and Mac OS. Most platforms can be described as a combination of the operating system and underlying hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

The Java platform has two components:

- The *Java Virtual Machine*
- The *Java Application Programming Interface (API)*

4.2.9 JAVA VIRTUAL MACHINE

A **Java virtual machine (JVM)** is a virtual machine that can execute Java byte code. It is the code execution component of the Java software platform. A Java virtual machine is a program which executes certain other programs, namely those containing Java byte code instructions.

JVMs are most often implemented to run on an existing operating system, but can also be implemented to run directly on hardware. A JVM provides an environment in which Java byte code can be executed, enabling such features as automated exception handling, which provides root-cause debugging information for every software error (exception), independent of the source code. A JVM is distributed along with a set of standard class libraries that implement the Java application programming interface (API). These libraries, bundled together with the JVM, form the Java Runtime Environment (JRE). JVMs are available for many hardware and software platforms. The use of the same byte code for all JVMs on all platforms allows Java to be described as a writeonce, runanywhere programming language, versus writeonce, compileanywhere, which describes cross-platform compiled languages. Thus, the JVM is a crucial component of the Java platform.

4.2.10 JAVA API

The API is a large collection of ready-made software components that provide many useful capabilities. It is grouped into libraries of related classes and interfaces; these libraries are known as *packages*.

As a platform-independent environment, the Java platform can be a bit slower than native

code. However, advances in compiler and virtual machine technologies are bringing performance close to that of native code without threatening portability.

4.3 BACK END

4.3.1 WHAT IS MYSQL?

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. The MySQL Web site (<http://www.mysql.com/>) provides the latest information about MySQL software.

- MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by MySQL AB. MySQL AB is a commercial company, founded in 1995 by the MySQL developers. It is a second generation Open Source company that unites Open Source values and methodology with a successful business model.
- The MySQL® software delivers a very fast, multi-threaded, multi-user, and robust SQL (Structured Query Language) database server. MySQL Server is intended for mission-critical, heavy-load production systems as well as for embedding into mass-deployed software. MySQL is a registered trademark of MySQL AB.
- The MySQL software is Dual Licensed. Users can choose to use the MySQL software as an Open Source product under the terms of the GNU General Public License or can purchase a standard commercial license from MySQL AB.

The company name was given after co-founder Monty Widenius's daughter, My, the SQL is "Structured Query Language.", AB is Swedish for limited partnership.

• MYSQL IS A DATABASE MANAGEMENT SYSTEM.

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

• MySQL databases are relational. What is MySQL?

A relational database stores data in separate tables rather than putting all the data in one

big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and “pointers” between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data. The SQL part of “MySQL” stands for “Structured Query Language”. SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax. SQL is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, “SQL-92” refers to the standard released in 1992, “SQL:1999” refers to the standard released in 1999, and “SQL:2003” refers to the current version of the standard. We use the phrase “the SQL standard” to mean the current version of the SQL Standard at any time

MySQL Overview

What does it do?

- MySQL is a database management system.

A database is a structured collection of data. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server.

- **MySQL is a relational database management system.**

A relational database stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The SQL part of “MySQL” stands for “Structured Query Language.”

- **MySQL software is Open Source.**

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. If you want to redistribute altered versions MySQL, other licences are available

- **MySQL Server works in client/server or embedded systems.**

The MySQL Database Software is a client/server system that consists of a multi-threaded SQL

server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

- **A large amount of contributed MySQL software is available.**

MySQL is a widely supported database, and very likely that most of your applications supports it.

MySQL Architecture

Three layer model:

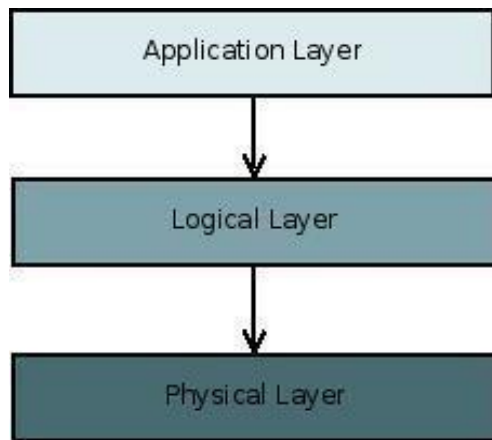


Fig 5.1 MySQL Architecture

- The application layer contains common network services for connection handling, authentication and security. This layer is where different clients interact with MySQL these clients can be written in different API's: .NET, Java, C, C++, PHP, Python, Ruby, Tcl, Eiffel, etc...
- The Logical Layer is where the MySQL intelligence resides, it includes functionality for query parsing, analysis, caching and all built-in functions (math, date...). This layer also provides functionality common across the storage engines.
- The Physical Layer is responsible for storing and retrieving all data stored in "MySQL". Associated with this layer are the storage engines, which MySQL interacts with very basic standard API's. Each storage engine has its strengths and weaknesses, some of these engines are MyISAM, InnoDB, CSV, NDB Cluster, Falcon, etc...

SOME INTERNAL COMPONENTS:

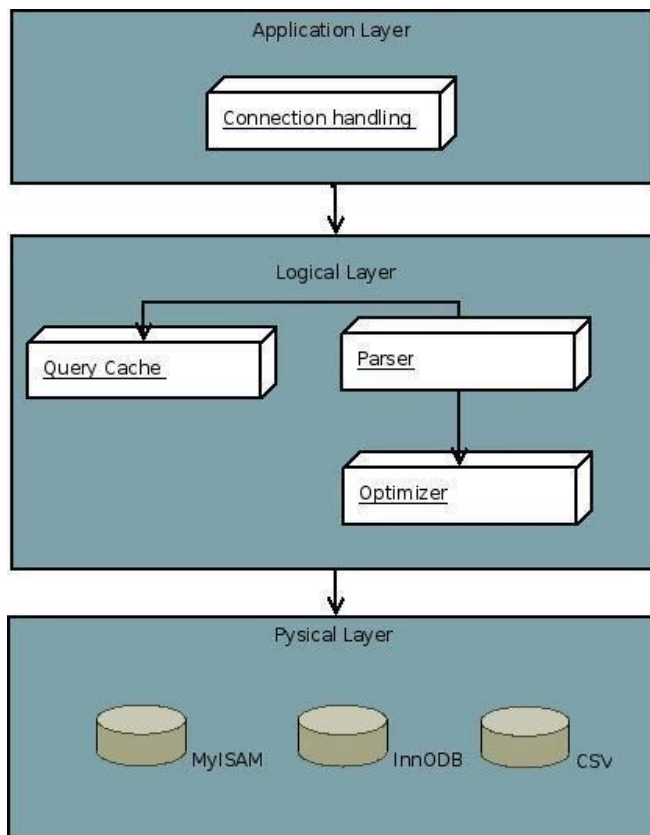


Fig 5.2 MySQL internal Components

Each client connection gets its own thread within the server process.

When clients (applications) connect to the MySQL server, the server needs to authenticate them.

Before even parsing the query, though, the server consults the query cache, which only stores SELECT statements, along with their result sets.

The storage engine does affect how the server optimizes query.

Storage engines:

MySQL supports several storage engines that act as handlers for different table types. MySQL storage engines include both those that handle transaction-safe tables and those that handle non-transaction-safe tables.

Feature	MyISAM	Memory	InnoDB	Archive	NDB
Storage limits	256TB	RAM	64TB	None	384EB
Transactions	No	No	Yes	No	Yes
Locking granularity	Table	Table	Row	Row	Row

MySQL Files:

In Linux the configuration file is typically located at `/etc/mysql/my.cnf`, but may vary for the different Linux flavours, this also applies to the database files themselves which are located in the

/var/lib/MySQL.

Regardless of the storage engine, every MySQL table you create is represented, on disk, by a .frm file, which describes the table's format (i.e. the table definition). The file bears the same name as the table, with a .frm extension. The .frm format is the same on all platforms but in the description of the .frm format that follows, the examples come from tables created under the Linux operating system.

```
/var/lib/mysql/db.frm    #Table definition
/var/lib/mysql/db.MYD    #MyISAM data file
/var/lib/mysql/db.MYI    #MyISAM Index file
/var/lib/mysql/ibdata1   #InnoDB data file
```

MySQL Cluster Overview

MySQL Cluster is a technology that enables clustering of in-memory databases in a shared-nothing system. The shared-nothing architecture enables the system to work with very inexpensive hardware, and with a minimum of specific requirements for hardware or software.

MySQL Cluster is designed not to have any single point of failure. In a shared-nothing system, each component is expected to have its own memory and disk, and the use of shared storage mechanisms such as network shares, network file systems, and SANs is not recommended or supported.

MySQL Cluster integrates the standard MySQL server with an in-memory clustered storage engine called NDB (which stands for "Network DataBase"). In our documentation, the term NDB refers to the part of the setup that is specific to the storage engine, whereas "MySQL Cluster" refers to the combination of one or more MySQL servers with the NDB storage engine. A MySQL Cluster consists of a set of computers, known as hosts, each running one or more processes. These processes, known as nodes, may include MySQL servers (for access to NDB data), data nodes (for storage of the data), one or more management servers, and possibly other specialized data access programs. The relationship of these components in a MySQL Cluster is shown here:

4.4 MODULES

- A modular design reduces complexity, facilitates change (a critical aspect of software maintainability), and results in easier implementation by encouraging parallel development of different part of system. Software with effective modularity is easier to develop because function may be compartmentalized and interfaces are simplified. Software architecture embodies modularity that is software is

divided into separately named and addressable components called modules that are integrated to satisfy problem requirements.

- Modularity is the single attribute of software that allows a program to be intellectually manageable. The five important criteria that enable us to evaluate a design method with respect to its ability to define an effective modular design are: Modular decomposability, Modular Comps ability, Modular Understandability, Modular continuity, Modular Protection.
- The following are the modules of the project, which is planned in aid to complete the project with respect to the proposed system, while overcoming existing system and also providing the support for the future enhancement.
- **LOGIN**

In this module, both the user and the admin will register in a centralized server. Admin will add the news along with keyword based segmentation in the server database. User will view the news information from the server from both the social media extraction as well as general News feeds. Admin plays a dual input role both from the social media as well as general News feeds.

- **SERVER**

In this module, all the data from both social media and general News feeds are stored in this centralized server for any data processing and filtering. Data classification with categorization will effectively happen in the server. The server will do all the data analysis and extraction of keywords and categorization will happen effectively in the server.

- **DATA CLASSIFICATION**

We apply stemming algorithm to remove the stop words from the Data. Stop words is the set of grammatical words or verbs or adverbs which adds meaning to the sentences in the Data. the purpose of removal of the stemming words is to extract important words from the Data. Once the important words are extracted from the Data we need to compare with the preloaded keywords for the Data Categorization. We need to train a dataset with respect

to the Data Categorization and classification. We need to compare and map the most related and repeated keywords present in the Data and compare with the total number of important keywords. This system will identify the Data Categorization and classification. We also extract Hash tagging keywords and compare with the Keywords from the social media for the data categorization.

- **AUTO ALERT SYSTEM**

In this module, user can feed the data to be alerted automatically even it is notified by the user. Once the news is published either in the social media or in the news feed automatically extracts the data is identified by the keyword comparison and automatic information is notified to the user. This model is very helpful to the user to track and alert any information even the data is missed out.

4.5 SYSTEM TESTING

Generally testing process is carried out to ensure that the system has been developed according to the required specifications and the expected output is properly obtained. There are two main categories of testing namely, the White Box Testing and the Black Box Testing. Each of this testing in turn consists of many types of testing.

TAXONOMY OF TESTING

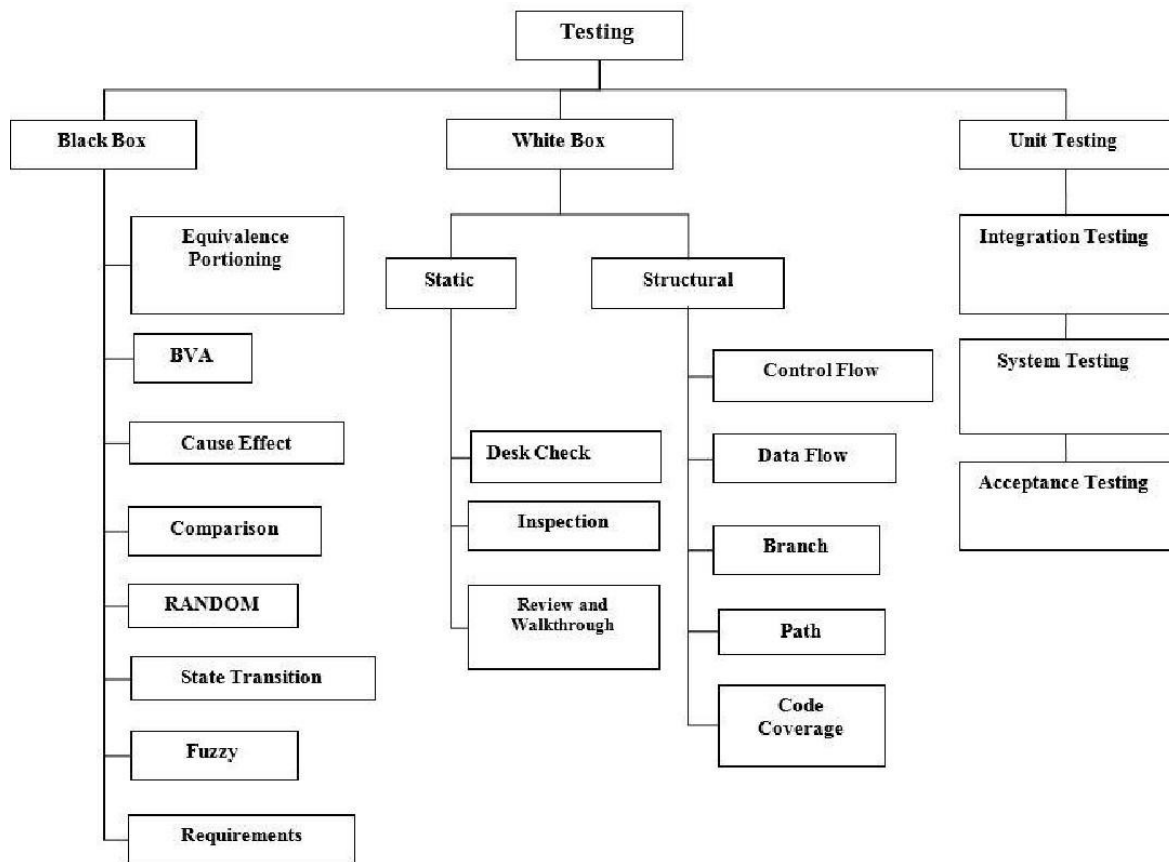


Fig 7.1 Taxonomy of Testing

WHITE BOX TESTING

White Box testing is also known as glass box testing. This type of testing, tests the internal structure of the program. This can be applied at the unit, integration and system levels of testing. Mostly, it is used in the unit level of the software testing process. Sometimes it may not reveal defects in areas which have not been implemented. It has its own advantages and its own disadvantages. The advantage is that knowing the programming language code and familiarizing with them may prove vital and help in identifying the errors quickly and at times may help in avoiding them at the earliest.

BLACK BOX TESTING

It is amongst the two methods of mostly used testing methods. This tests the main functionality of the program. It can be applied to every level of testing such as Unit, Integration, System and Acceptance levels of testing. Exhaustive input testing is required to find all errors. For doing this type of testing knowing the internal code and how it works is not needed but what it is supposed to do is known by the person who is performing the test. The test cases are developed based on the specific requirements according to the goals. There are Boundary Value Analysis, Class Partitioning, Cause Effect Graph etc.,

➤ □ UNIT TESTING:

Unit testing is also known as Module Testing which focuses on verification efforts on the module. The module is tested separately and this is carried out at the programming stage itself. Unit test comprises of the set of tests performed by an individual programmer before integration of the unit into the system. This will help to test each and every single part or we can say as each and every module completely.

These may even be small parts of the code and test cases will be developed which are independent of each other.

➤ □ FUNCTIONAL TESTING:

Functional testing is mainly used as a Quality Assurance process. This is a very simple process where each function is provided with an appropriate input and is verified against an expected output and with boundary values. This would help in ensuring that the output is as acquired according to the expectations and would help assuring the quality. The various functions developed using Java are separately tested for their proper working by executing them as separate files.

➤ □ INTEGRATION TESTING:

It is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with in the interface. It takes the unit tested modules and builds a program structure. Integration of all the components to form the entire system and an overall testing is executed. Integration testing is any type of

software testing that seeks to verify the interfaces between components against a software design. Software components may be integrated in an iterative way or all together as in a on the whole approach. Normally the former is considered a better practice since it allows interface issues to be located more quickly and fixed. It is mainly done based on by taking into account of the number of modules used, how many number of interfaces maybe required to integrate them, which had to be combined and clustering process.

➤ □ **VALIDATION TESTING:**

Validation test succeeds when the software functions in a manner that can be reasonably expected by the client. Software validation is achieved through a series of black box testing which confirms to the requirements. The software is validated based on the series of tests that it passes through according to the condition posed by the customer. Mostly the customer main requirements would be to make every process as simple as possible and to reduce the complexity of the usage of the final product. Taking all these conditions into mind the validation testing is done and the various test cases are design.

➤ □ **SYSTEM TESTING:**

System testing of software or hardware is testing conducted on complete, integrated system to evaluate the system's compliance with its specified requirements. Once all of the modules have been completely developed and both unit testing and integration testing is done on the various parts of the modules later the system testing is done so as to ensure that the requirements are fulfilled properly. All it basically does is it performs tests to find the discrepancies between the system and its original objective, current specifications and system documentation. If any discrepancy is to be found the respective errors will be rectified and again system testing will be performed to make sure the rectification does not introduce a new error into the system.

➤ □ **STRUCTURE TESTING:**

It is concerned with exercising the internal logic of a program and traversing certain execution paths. Structure testing takes into account of all process that works internally to make the entire system to work properly. The basic structure and the background codes and fragments that help in upholding the system are used.

This goes layer by layer as in till reaching the core process. The various layers or levels depends on the type of the project and the domain it comes under. This is considered to be a part

of the White Box Testing Process. This is done so as to make sure all the internal processes work properly. If They don't then the probability that the entire process may collapse is a possibility and this causes a grave danger to the project leading to failure.

TESTING IN PARTICULAR

UNIT TESTING

Unit testing is a software testing method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use.

In this project, all statements are executed properly. All units of program programs are tested in different computer. And the result of the project is same in all system.

INTEGRATION TESTING

Integration testing (sometimes called **integration and testing**, abbreviated **I&T**) is the phase in software testing in which individual software modules are combined and tested as a group. It occurs after unit testing and before validation testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

CHAPTER 8

RESULT AND DISCUSSION

In this project we extracted the keywords from the news articles by removing the stop words using stemming algorithm and categorised the news articles using those keywords. Also, the news will be visible to the user on a priority basis i.e. the news of a particular category user views the most & likes will be on top followed by the other news news posted by the admin. Also, if the user notification likes a news any news of that particular category posted by the admin in future will trigger a notification mail from the admin to the user. User can also easily filter the content & find the news of his interest very easily.

CHAPTER 9

CONCLUSION AND FUTURE WORK

CONCLUSION:

Thus the paper infers that user will get the news which is mostly liked by the public. Through this system we implement a new prototype news classification and also in social media. The News are categorized and classified with respect to the category classified based on the set of keywords assigned to the centralized server. All the Keywords are trained in our Server using Machine Learning algorithm to principally classify the news according to the category. We also apply Stemming algorithm to remove all the unwanted keywords and to filter only the main and important keywords. We assign the keywords like Ball, Bat, Dhoni, Virat Kohli, Ground and other related words to the Sports Category, same way rest of the keywords are assigned to the corresponding categories. We assign synonyms into this category which is very will be effective in the proper segregation of the messages. We also add Automatic Alert system in this Application about a particular Keyword or its synonyms so that automatic notification is intimated to the customer even he / she misses out to view the News.

FUTURE WORK:

In the current Scenario, only the set of trained keywords are trained and compared, but in the future development process we can include synonyms and related set of keywords also. Once the admin provides one keywords our system should dynamically add the synonyms keywords associated with that Keyword. This process will increase the reliability of the project / Implementation.

APPENDIX

CODE:

User Login:

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{
    //GEN-FIRST:event_jButton1ActionPerformed
        // TODO add your handling code here:
        Implemts_dtls _implements=new Implemts_dtls();
        String uname=jTextField1.getText().toString().trim();
        String pass=jPasswordField1.getText().toString().trim();

        if(uname.equalsIgnoreCase("Admin")&&pass.equalsIgnoreCase("Admin")){
            this.dispose();
            new AdminMenu().setVisible(true);
        }
        else if(_implements.LoginValid(uname, pass)){
            this.dispose();
            new UserHome().setVisible(true);
        }
        else{
            JOptionPane.showMessageDialog(rootPane, "Invalid Login
            Credentials");
        }
    }
    //GEN-LAST:event_jButton1ActionPerformed

    private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
    {
        //GEN-FIRST:event_jButton3ActionPerformed
            // TODO add your handling code here:
```

```

        this.dispose();
        new UserRegister().setVisible(true);
    } //GEN-LAST:event_jButton3ActionPerformed

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the
default look and feel.
        * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
        */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(UserLogin.class.getName()).log(java.util.lo
gging.Level.SEVERE, null, ex);

        } catch (InstantiationException ex) {

```

```
java.util.logging.Logger.getLogger(UserLogin.class.getName()).log(java.util.lo  
gging.Level.SEVERE, null, ex);
```

```
    } catch (IllegalAccessException ex) {
```

```
java.util.logging.Logger.getLogger(UserLogin.class.getName()).log(java.util.lo  
gging.Level.SEVERE, null, ex);
```

```
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(UserLogin.class.getName()).log(java.util.lo  
gging.Level.SEVERE, null, ex);
```

```
    }
```

```
//</editor-fold>
```

```
/* Create and display the form */
```

```
java.awt.EventQueue.invokeLater(new Runnable() {
```

```
    public void run() {
```

```
        new UserLogin().setVisible(true);
```

```
    }
```

```
});
```

```
}
```

Home page:

```
public class UserHome extends javax.swing.JFrame {
```

```
    /**
```

```
     * Creates new form UserHome
```

```
    */
```

```
    Implemts_dtls _imple=new Implemts_dtls();
```

```
    public UserHome() {
```

```
        String tokenizer="";
```

```
        initComponents();
```

```
        String dbdata=_imple.GetAllNews();
```

```
        StringTokenizer st=new StringTokenizer(dbdata,"$");
```

```
        while(st.hasMoreTokens()){
```

```
            tokenizer+=st.nextToken()+"\n";
```

```
        }
```

```
        JTextArea1.setText(tokenizer);
```

```
    }
```

```
    /**
```

```
     * This method is called from within the constructor to initialize the form.
```

```
     * WARNING: Do NOT modify this code. The content of this method is  
always
```

```
     * regenerated by the Form Editor.
```

```
    */
```

```
    @SuppressWarnings("unchecked")
```

```
    // <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-  
BEGIN: initComponents
```

```
    private void initComponents() {
```

```

jLabel1 = new javax.swing.JLabel();
jLabel2 = new javax.swing.JLabel();
jScrollPane1 = new javax.swing.JScrollPane();
jTextArea1 = new javax.swing.JTextArea();
jButton1 = new javax.swing.JButton();
jButton2 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE)
;

jLabel1.setFont(new java.awt.Font("Times New Roman", 1, 18)); //
NOI18N
jLabel1.setText("USER HOME");

jLabel2.setFont(new java.awt.Font("Times New Roman", 1, 18)); //
NOI18N
jLabel2.setText("NEWS");

jTextArea1.setColumns(20);
jTextArea1.setRows(5);
jScrollPane1.setViewportView(jTextArea1);

jButton1.setFont(new java.awt.Font("Times New Roman", 1, 18)); //
NOI18N
jButton1.setText("LIKE");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        jButton1ActionPerformed(evt);
    }
});

jButton2.setFont(new java.awt.Font("Times New Roman", 1, 18)); //
NOI18N
jButton2.setText("BACK");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addGap(38, 38, 38)
        .addComponent(jLabel2)
        .addGap(56, 56, 56)

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
EADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(38, 38, 38)
                .addComponent(jLabel2)
                .addGap(56, 56, 56)

```

EADING)

```
        .addComponent(jScrollPane1,  
javax.swing.GroupLayout.PREFERRED_SIZE, 397,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,  
layout.createSequentialGroup()  
            .addComponent(jButton1)  
            .addGap(70, 70, 70)  
            .addComponent(jButton2)  
            .addGap(76, 76, 76))))  
        .addGroup(layout.createSequentialGroup()  
            .addGap(178, 178, 178)  
            .addComponent(jLabel1)))  
        .addContainerGap(57, Short.MAX_VALUE))  
    );  
    layout.setVerticalGroup(
```

```
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
        .addGroup(layout.createSequentialGroup()
```

```
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.L  
EADING)
```

```
            .addGroup(layout.createSequentialGroup()  
                .addGap(28, 28, 28)  
                .addComponent(jLabel1)  
                .addGap(33, 33, 33)  
                .addComponent(jScrollPane1,  
javax.swing.GroupLayout.PREFERRED_SIZE, 131,  
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```

        .addGroup(layout.createSequentialGroup()
            .addGap(129, 129, 129)
            .addComponent(jLabel2)))
        .addGap(29, 29, 29)

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.B
ASELINE)
        .addComponent(jButton1)
        .addComponent(jButton2)
        .addContainerGap(26, Short.MAX_VALUE))
    );

    pack();
} // </editor-fold> // GEN-END: initComponents

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_jButton1ActionPerformed
    // TODO add your handling code here:

    String select= jTextArea1.getSelectedText();
    String result=_imple.InsertReview(UserEntity.getname(), select);
    JOptionPane.showMessageDialog(rootPane, result);
} // GEN-LAST:event_jButton1ActionPerformed

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_jButton2ActionPerformed
    // TODO add your handling code here:
    this.dispose();
    new UserLogin().setVisible(true);
}

```

```

} //GEN-LAST:event_jButton2ActionPerformed

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the
default look and feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(UserHome.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);

    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(UserHome.class.getName()).log(java.util.l

```

```

        logging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(UserHome.class.getName()).log(java.util.l
logging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(UserHome.class.getName()).log(java.util.l
logging.Level.SEVERE, null, ex);
    }
//</editor-fold>

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        new UserHome().setVisible(true);
    }
});
}

```

Admin Post:

```

public class AdminPost extends javax.swing.JFrame {

    /**
     * Creates new form AdminPost
     */
    public AdminPost() {
        initComponents();
    }

```

```

}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-
BEGIN: initComponents
private void initComponents() {

    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jTextField1 = new javax.swing.JTextField();
    jButton1 = new javax.swing.JButton();
    jButton2 = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE)
    ;

    jLabel1.setFont(new java.awt.Font("Times New Roman", 1, 24)); //
NOI18N
    jLabel1.setText("Admin Post");

    jLabel2.setFont(new java.awt.Font("Times New Roman", 1, 18)); //
NOI18N

```

```
jLabel2.setText("News");
```

```
jTextField1.setFont(new java.awt.Font("Times New Roman", 1, 18)); //
NOI18N
```

```
jButton1.setFont(new java.awt.Font("Times New Roman", 1, 18)); //
NOI18N
```

```
jButton1.setText("POST");
jButton1.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton1ActionPerformed(evt);
    }
});
```

```
jButton2.setFont(new java.awt.Font("Times New Roman", 1, 18)); //
NOI18N
```

```
jButton2.setText("BACK");
jButton2.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton2ActionPerformed(evt);
    }
});
```

```
javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
```

```
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```

        .addGroup(layout.createSequentialGroup())

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
        .addGroup(layout.createSequentialGroup()
            .addGap(86, 86, 86)
            .addComponent(jButton1,
                javax.swing.GroupLayout.PREFERRED_SIZE, 162,
                javax.swing.GroupLayout.PREFERRED_SIZE)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
                javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(jButton2,
                javax.swing.GroupLayout.PREFERRED_SIZE, 211,
                javax.swing.GroupLayout.PREFERRED_SIZE))

        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(55, 55, 55)
                .addComponent(jLabel2,
                    javax.swing.GroupLayout.PREFERRED_SIZE, 70,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(57, 57, 57)
                .addComponent(jTextField1,
                    javax.swing.GroupLayout.PREFERRED_SIZE, 325,
                    javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(layout.createSequentialGroup()
                .addGap(251, 251, 251)

```

```

        .addComponent(jLabel1)))
    .addContainerGap(46, Short.MAX_VALUE)
);
layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addGap(62, 62, 62)
        .addComponent(jLabel1)
        .addGap(68, 68, 68)

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jLabel2,
javax.swing.GroupLayout.PREFERRED_SIZE, 48,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, 48,
javax.swing.GroupLayout.PREFERRED_SIZE))

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED
, 35, Short.MAX_VALUE)

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jButton1,
javax.swing.GroupLayout.PREFERRED_SIZE, 49,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(jButton2,

```

```

javax.swing.GroupLayout.PREFERRED_SIZE, 49,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGap(67, 67, 67))
    );

    pack();
} // </editor-fold> // GEN-END: initComponents

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_jButton2ActionPerformed
    // TODO add your handling code here:
    this.dispose();
    new AdminMenu().setVisible(true);
} // GEN-LAST: event_jButton2ActionPerformed

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_jButton1ActionPerformed
    // TODO add your handling code here:
    String News = jTextField1.getText().toString().trim();
    Implements_dtls user_Dtls = new Implements_dtls();
    int i = user_Dtls.PostKeywords_details(News);
    if (i > 0) {
        JOptionPane.showMessageDialog(rootPane, "Details saved");
    } else {
        JOptionPane.showMessageDialog(rootPane, "Details Not saved");
    }
}

} // GEN-LAST: event_jButton1ActionPerformed

```

```

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the
default look and feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(AdminPost.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(AdminPost.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

```

```
java.util.logging.Logger.getLogger(AdminPost.class.getName()).log(java.util.l  
ogging.Level.SEVERE, null, ex);
```

```
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(AdminPost.class.getName()).log(java.util.l  
ogging.Level.SEVERE, null, ex);
```

```
    }
```

```
//</editor-fold>
```

```
/* Create and display the form */
```

```
java.awt.EventQueue.invokeLater(new Runnable() {
```

```
    public void run() {
```

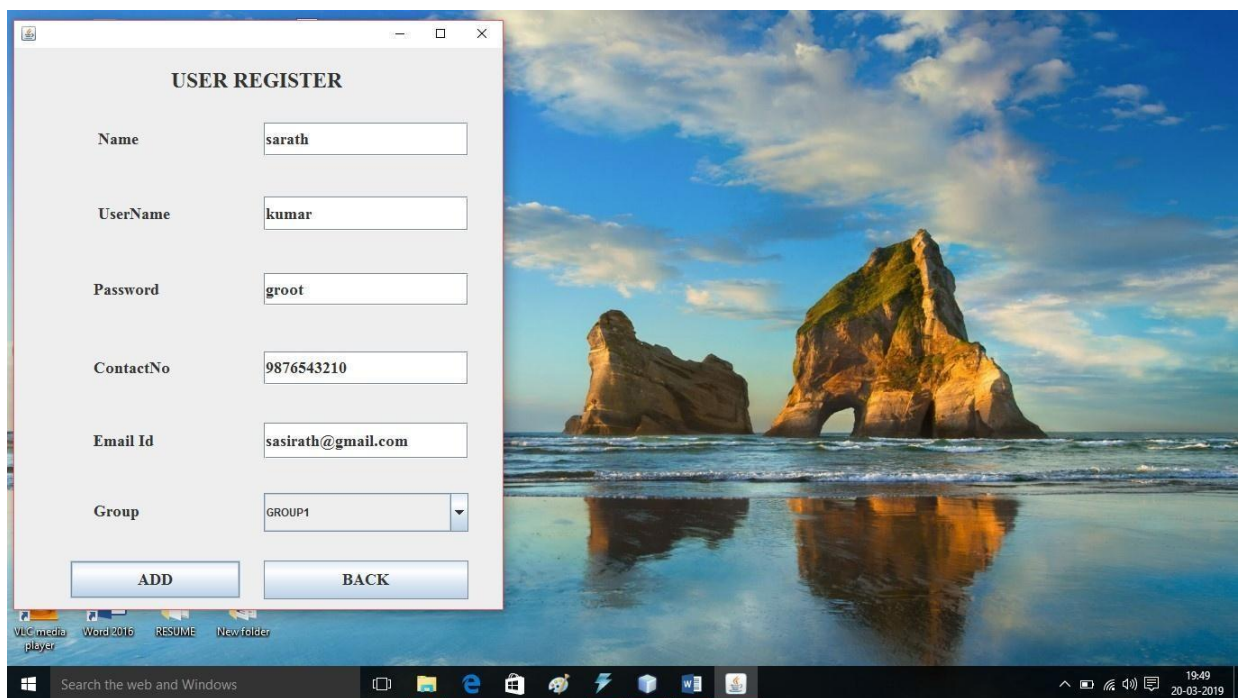
```
        new AdminPost().setVisible(true);
```

```
    }
```

```
});
```

```
}
```

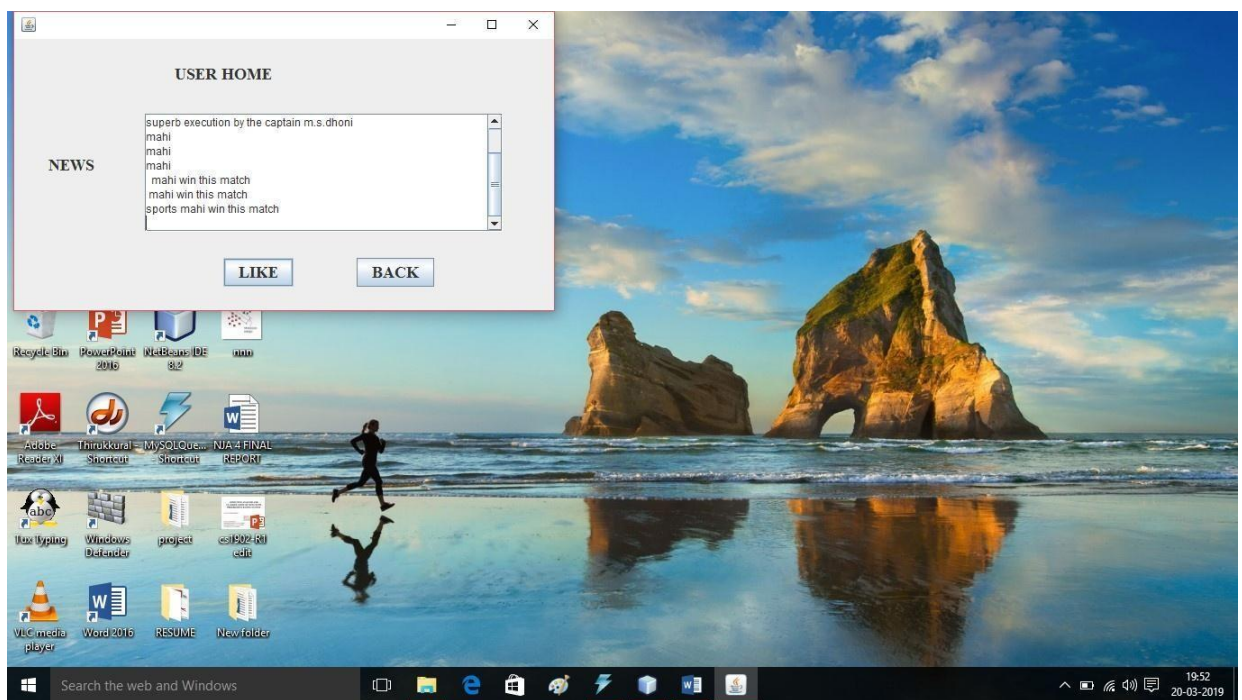
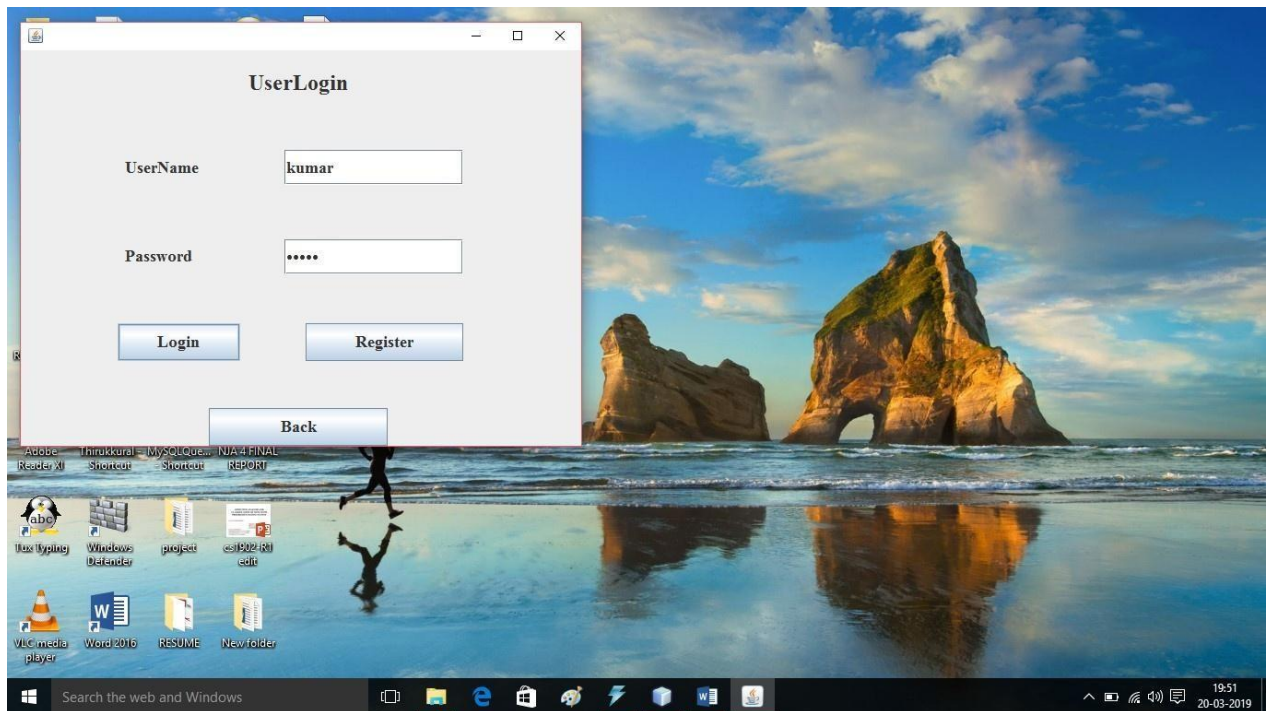
SCREENSHOT:

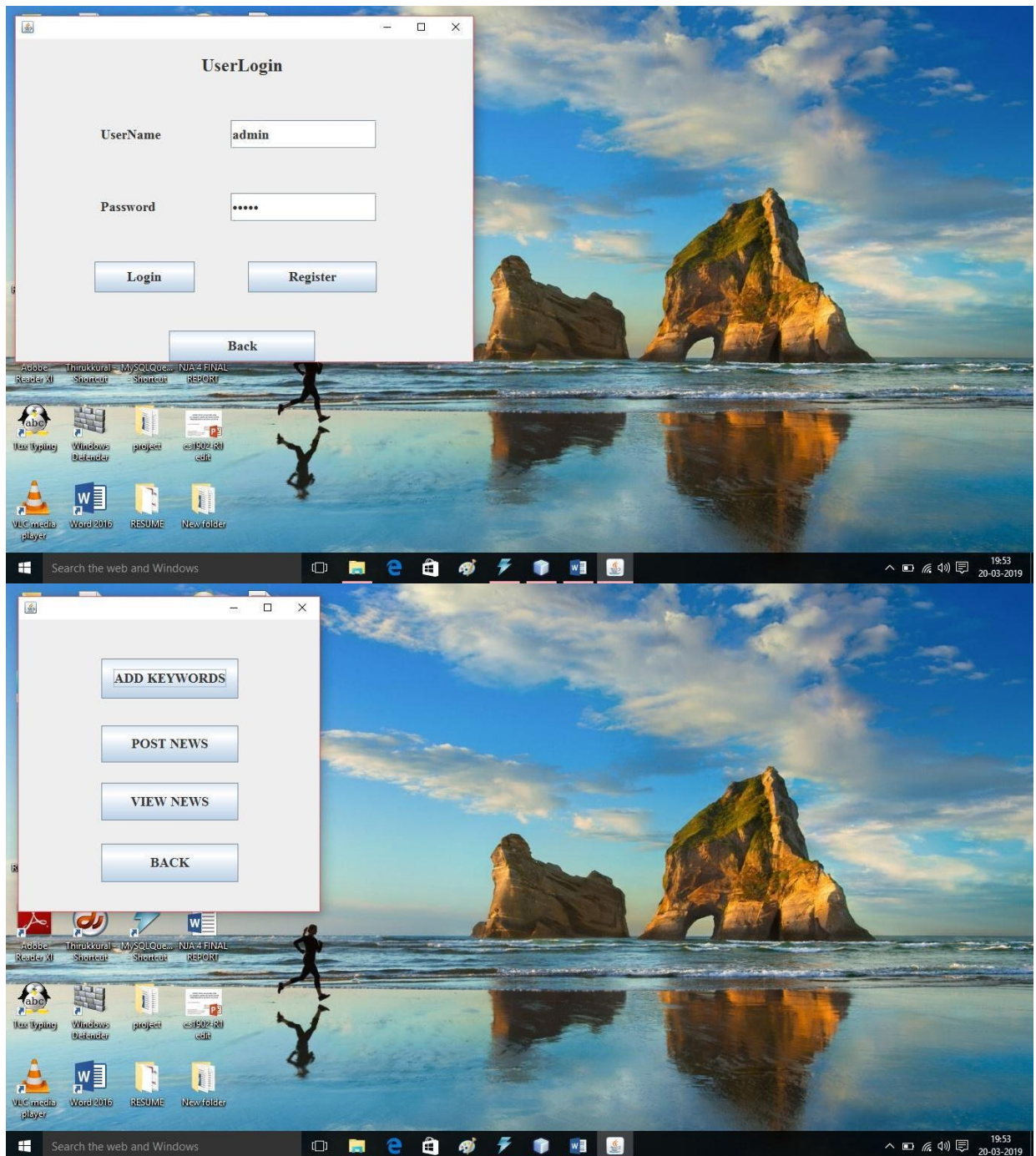


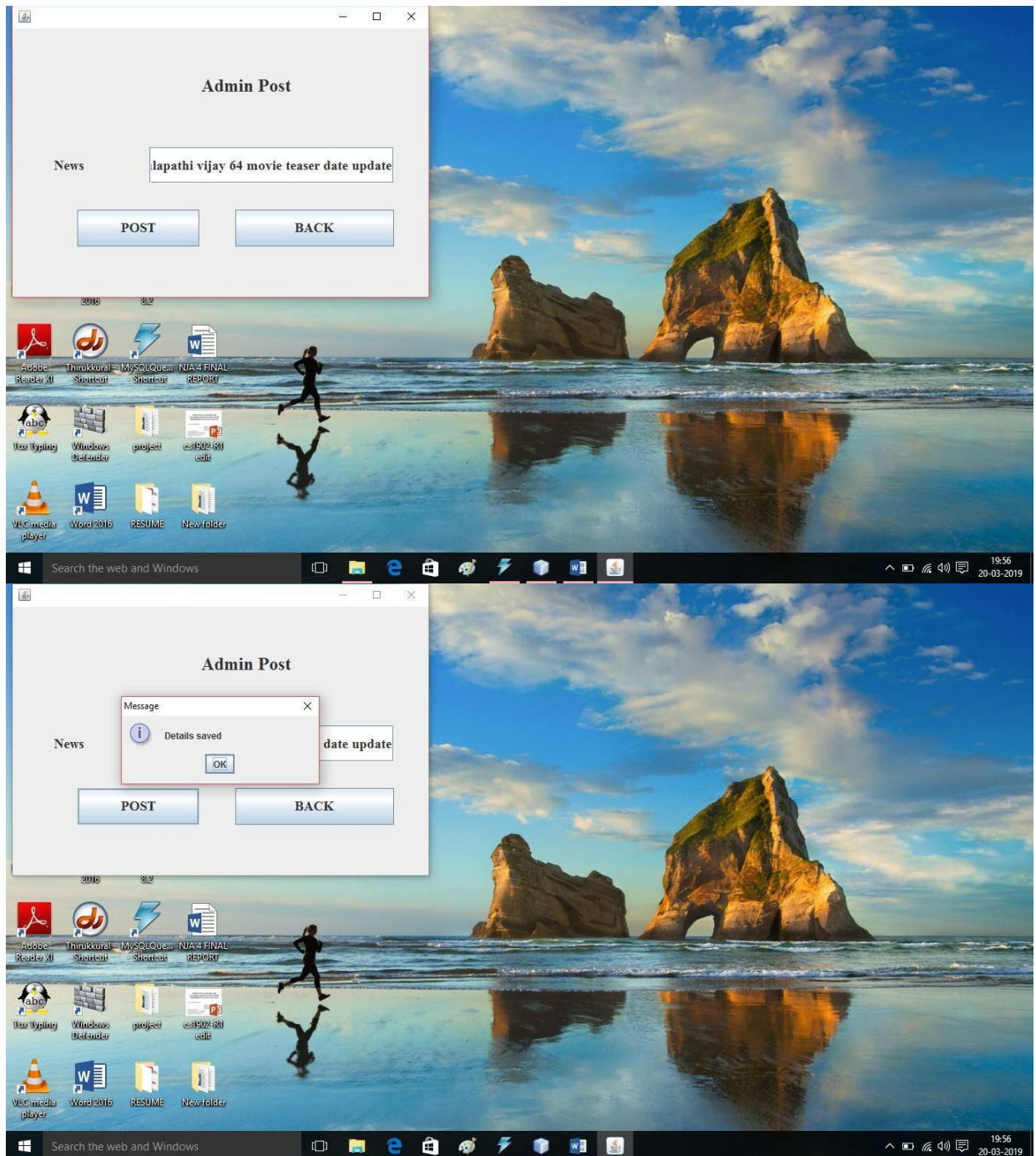
The screenshot displays a Windows desktop environment. In the foreground, a window titled "USER REGISTER" is open, featuring a registration form with the following fields and values:

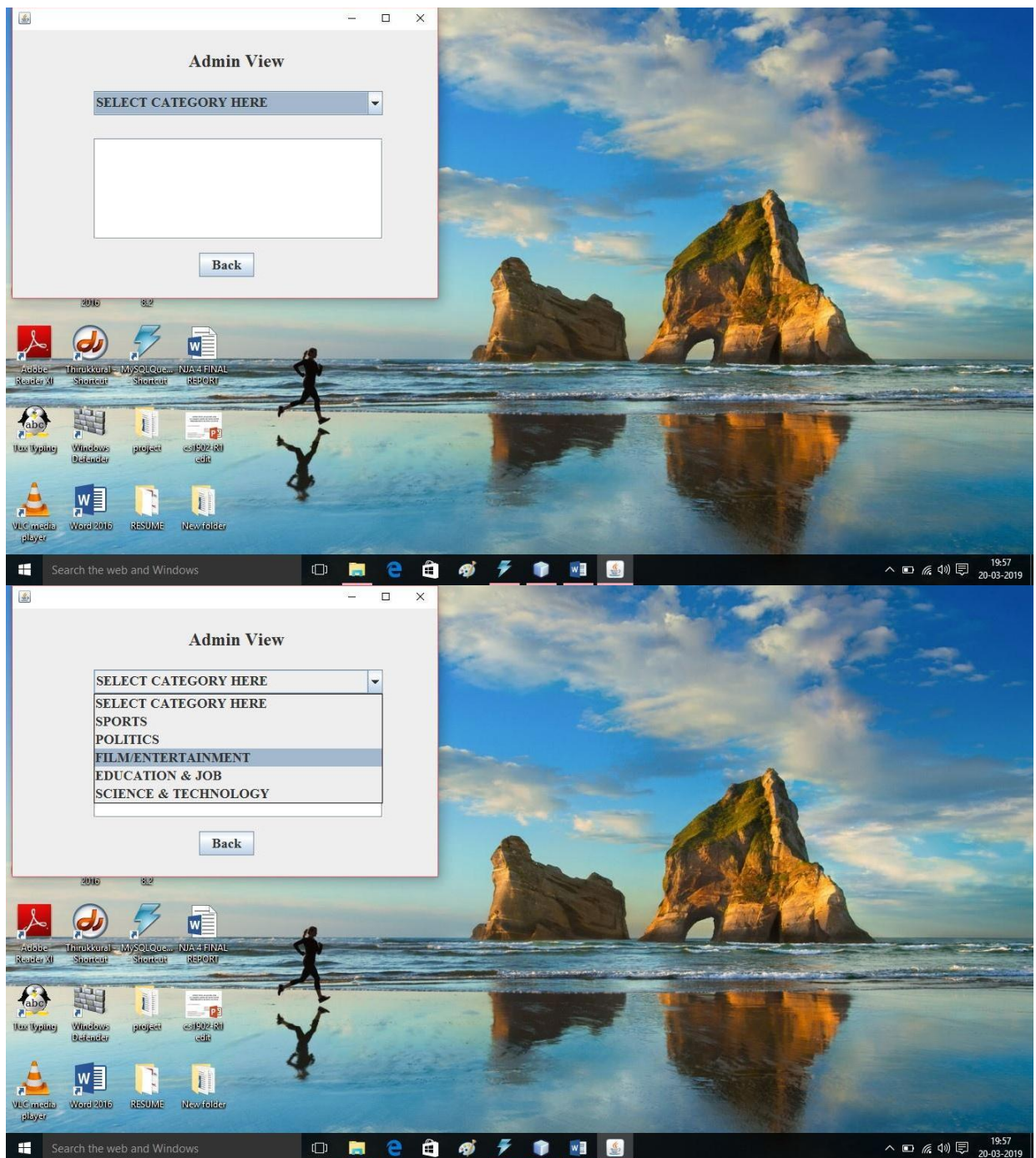
Field	Value
Name	sarath
UserName	kumar
Password	groot
ContactNo	9876543210
Email Id	sasirath@gmail.com
Group	GROUP1

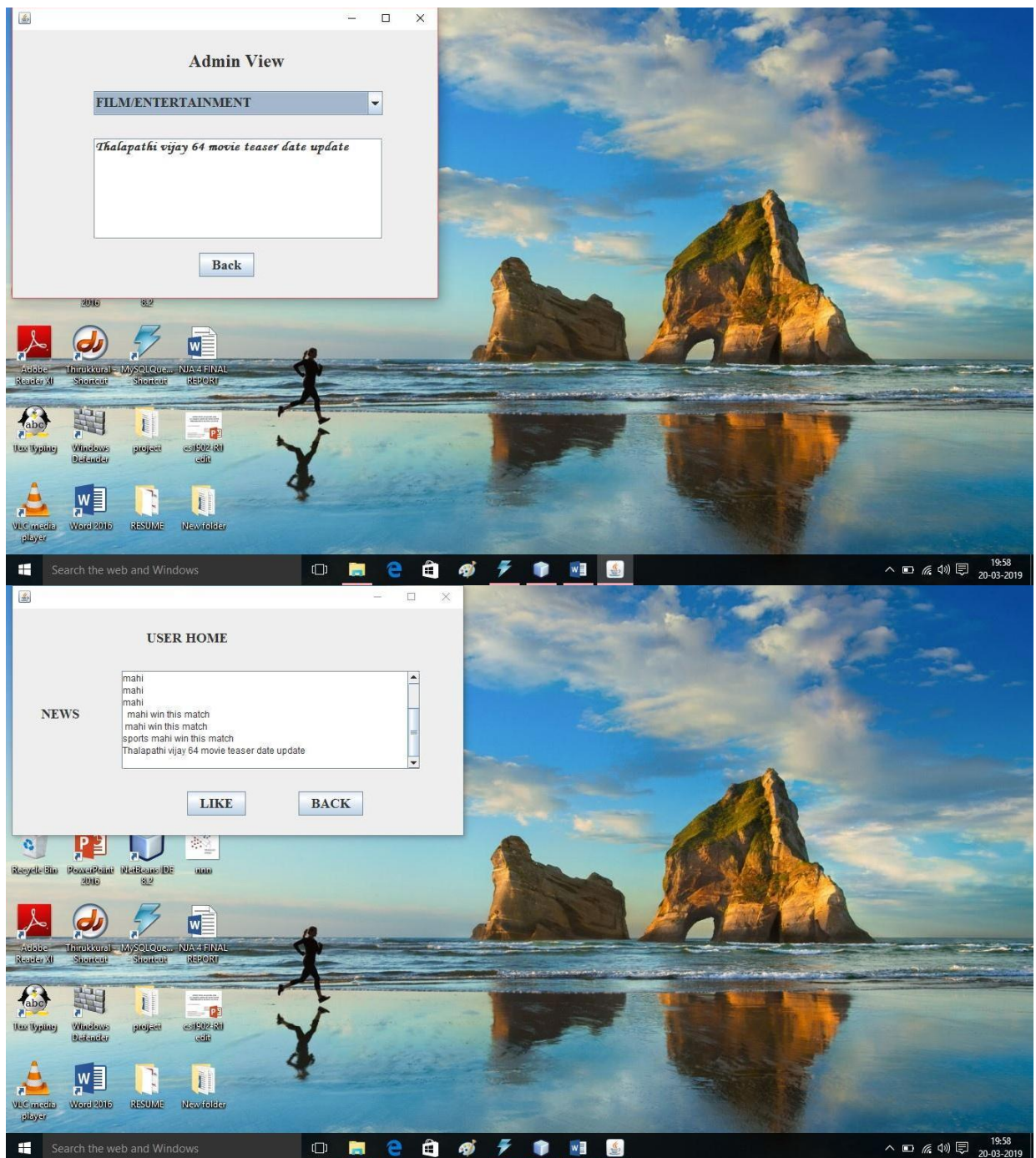
At the bottom of the form are two buttons: "ADD" and "BACK". The desktop background is a scenic image of a rocky coastline with waves and a large rock formation. The taskbar at the bottom shows the Windows logo, a search bar, and several application icons. The system tray on the right indicates the time as 19:49 on 20-03-2019.











REFERENCES

- [1] T. Rosenstiel, *Twitter and the News: How People use the Social Network to Learn About the World*. Arlington, VA, USA: American Press Institute, 2015.
- [2] Z. Ma, A. Sun, Q. Yuan, and G. Cong, "Tagging your tweets: A probabilistic modeling of hashtag annotation in twitter," in *Proc. 23rd ACM Int. Conf. Conf. Inform. Knowl. Manage.*, 2014, pp. 999–1008.
- [3] T.-A. Hoang-Vu, A. Bessa, L. Barbosa, and J. Freire, "Bridging vocabularies to link tweets and news," in *Proc. Int. Workshop Web Databases*, 2014.
- [4] Y. Gong, Q. Zhang, and X. Huang, "Hashtag recommendation using dirichlet process mixture models incorporating types of hashtags," in *Proc. Empirical Methods Natural Language Process.*, 2015, pp. 401–410.
- [5] Y. Gong and Q. Zhang, "Hashtag recommendation using attention-based convolutional neural network," in *Proc. 25th Int. Joint Conf. Artif. Intell.*, 2016, pp. 2782–2788.
- [6] B. Shi, G. Ifrim, and N. Hurley, "Learning-to-rank for real-time high-precision hashtag recommendation for streaming news," in *Proc. 25th Int. Conf. World Wide Web*, 2016, pp. 1191–1202.
- [7] K. S. Hasan and V. Ng, "Automatic keyphrase extraction: A survey of the state of the art," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics*, 2014, pp. 1262–1273.
- [8] T.-Y. Liu, "Learning to rank for information retrieval," *Foundations Trends Inform. Retrieval*, vol. 3, pp. 225–331, 2009.
- [9] R. Dovgopoi and M. Nohelty, "Twitter hash tag recommendation," *CoRR*, vol. abs/1502.00094, 2015, <http://arxiv.org/abs/1502.00094>

[10] A. Mazzia and J. Juett, "Suggesting hashtags on twitter," EECS 545 Project, 2011, <http://www-personal.umich.edu/~amazzia/pubs/545-final.pdf>