

Mobile Intelligence Assisted by Data Analytics and Cognitive Computing 2021

Lead Guest Editor: Yin Zhang

Guest Editors: Huimin Lu and Haider Abbas





**Mobile Intelligence Assisted by Data Analytics
and Cognitive Computing 2021**

Wireless Communications and Mobile Computing

**Mobile Intelligence Assisted by Data
Analytics and Cognitive Computing
2021**

Lead Guest Editor: Yin Zhang


Guest Editors: Huimin Lu and Haider Abbas





Copyright © 2021 Hindawi Limited. All rights reserved.

This is a special issue published in “Wireless Communications and Mobile Computing.” All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Chief Editor

Zhipeng Cai , USA

Associate Editors

Ke Guan , China
Jaime Lloret , Spain
Maode Ma , Singapore

Academic Editors

Muhammad Inam Abbasi, Malaysia
Ghufran Ahmed , Pakistan
Hamza Mohammed Ridha Al-Khafaji ,
Iraq
Abdullah Alamoodi , Malaysia
Marica Amadeo, Italy
Sandhya Aneja, USA
Mohd Dilshad Ansari, India
Eva Antonino-Daviu , Spain
Mehmet Emin Aydin, United Kingdom
Parameshchhari B. D. , India
Kalapaveen Bagadi , India
Ashish Bagwari , India
Dr. Abdul Basit , Pakistan
Alessandro Bazzi , Italy
Zdenek Becvar , Czech Republic
Nabil Benamar , Morocco
Olivier Berder, France
Petros S. Bithas, Greece
Dario Bruneo , Italy
Jun Cai, Canada
Xuesong Cai, Denmark
Gerardo Canfora , Italy
Rolando Carrasco, United Kingdom
Vicente Casares-Giner , Spain
Brijesh Chaurasia, India
Lin Chen , France
Xianfu Chen , Finland
Hui Cheng , United Kingdom
Hsin-Hung Cho, Taiwan
Ernestina Cianca , Italy
Marta Cimitile , Italy
Riccardo Colella , Italy
Mario Collotta , Italy
Massimo Condoluci , Sweden
Antonino Crivello , Italy
Antonio De Domenico , France
Floriano De Rango , Italy


Antonio De la Oliva , Spain
Margot Deruyck, Belgium
Liang Dong , USA
Praveen Kumar Donta, Austria
Zhuojun Duan, USA
Mohammed El-Hajjar , United Kingdom
Oscar Esparza , Spain
Maria Fazio , Italy
Mauro Femminella , Italy
Manuel Fernandez-Veiga , Spain
Gianluigi Ferrari , Italy
Luca Foschini , Italy
Alexandros G. Fragkiadakis , Greece
Ivan Ganchev , Bulgaria
Óscar García, Spain
Manuel García Sánchez , Spain
L. J. García Villalba , Spain
Miguel Garcia-Pineda , Spain
Piedad Garrido , Spain
Michele Girolami, Italy
Mariusz Glabowski , Poland
Carles Gomez , Spain
Antonio Guerrieri , Italy
Barbara Guidi , Italy
Rami Hamdi, Qatar
Tao Han, USA
Sherief Hashima , Egypt
Mahmoud Hassaballah , Egypt
Yejun He , China
Yixin He, China
Andrej Hrovat , Slovenia
Chunqiang Hu , China
Xuexian Hu , China
Zhenghua Huang , China
Xiaohong Jiang , Japan
Vicente Julian , Spain
Rajesh Kaluri , India
Dimitrios Katsaros, Greece
Muhammad Asghar Khan, Pakistan
Rahim Khan , Pakistan
Ahmed Khattab, Egypt
Hasan Ali Khattak, Pakistan
Mario Kolberg , United Kingdom
Meet Kumari, India
Wen-Cheng Lai , Taiwan

Jose M. Lanza-Gutierrez, Spain
Pavlos I. Lazaridis , United Kingdom
Kim-Hung Le , Vietnam
Tuan Anh Le , United Kingdom
Xianfu Lei, China
Jianfeng Li , China
Xiangxue Li , China
Yaguang Lin , China
Zhi Lin , China
Liu Liu , China
Mingqian Liu , China
Zhi Liu, Japan
Miguel López-Benítez , United Kingdom
Chuanwen Luo , China
Lu Lv, China
Basem M. ElHalawany , Egypt
Imadeldin Mahgoub , USA
Rajesh Manoharan , India
Davide Mattera , Italy
Michael McGuire , Canada
Weizhi Meng , Denmark
Klaus Moessner , United Kingdom
Simone Morosi , Italy
Amrit Mukherjee, Czech Republic
Shahid Mumtaz , Portugal
Giovanni Nardini , Italy
Tuan M. Nguyen , Vietnam
Petros Nicolitidis , Greece
Rajendran Parthiban , Malaysia
Giovanni Pau , Italy
Matteo Petracca , Italy
Marco Picone , Italy
Daniele Pinchera , Italy
Giuseppe Piro , Italy
Javier Prieto , Spain
Umair Rafique, Finland
Maheswar Rajagopal , India
Sujan Rajbhandari , United Kingdom
Rajib Rana, Australia
Luca Reggiani , Italy
Daniel G. Reina , Spain
Bo Rong , Canada
Mangal Sain , Republic of Korea
Praneet Saurabh , India


Hans Schotten, Germany
Patrick Seeling , USA
Muhammad Shafiq , China
Zaffar Ahmed Shaikh , Pakistan
Vishal Sharma , United Kingdom
Kaize Shi , Australia
Chakchai So-In, Thailand
Enrique Stevens-Navarro , Mexico
Sangeetha Subbaraj , India
Tien-Wen Sung, Taiwan
Suhua Tang , Japan
Pan Tang , China
Pierre-Martin Tardif , Canada
Sreenath Reddy Thummaluru, India
Tran Trung Duy , Vietnam
Fan-Hsun Tseng, Taiwan
S Velliangiri , India
Quoc-Tuan Vien , United Kingdom
Enrico M. Vitucci , Italy
Shaohua Wan , China
Dawei Wang, China
Huaqun Wang , China
Pengfei Wang , China
Dapeng Wu , China
Huaming Wu , China
Ding Xu , China
YAN YAO , China
Jie Yang, USA
Long Yang , China
Qiang Ye , Canada
Changyan Yi , China
Ya-Ju Yu , Taiwan
Marat V. Yuldashev , Finland
Sherali Zeadally, USA
Hong-Hai Zhang, USA
Jiliang Zhang, China
Lei Zhang, Spain
Wence Zhang , China
Yushu Zhang, China
Kechen Zheng, China
Fuhui Zhou , USA
Meiling Zhu, United Kingdom
Zhengyu Zhu , China

Contents





Information Acquisition Incentive Mechanism Based on Evolutionary Game Theory

Lihong Dong , Xirong Wang , Beizhan Liu, Tianwei Zheng , and Zheng Wang
Research Article (11 pages), Article ID 5525791, Volume 2021 (2021)

Forensic Analysis of Social Networking Applications on an Android Smartphone

Anoshia Menahil, Waseem Iqbal , Mohsin Iftikhar, Waleed Bin Shahid, Khwaja Mansoor, and Saddam Rubab
Research Article (36 pages), Article ID 5567592, Volume 2021 (2021)

Reducing Entropy Overestimation in Soft Actor Critic Using Dual Policy Network

Hamid Ali , Hammad Majeed , Imran Usman , and Khaled A. Almejalli 
Research Article (13 pages), Article ID 9920591, Volume 2021 (2021)

Deep Personalized Medical Recommendations Based on the Integration of Rating Features and Review Sentiment Analysis

Qihang Zhou, Lei Su , Liping Wu, and Di Jiang
Research Article (9 pages), Article ID 5551318, Volume 2021 (2021)

Research Article

Information Acquisition Incentive Mechanism Based on Evolutionary Game Theory

Lihong Dong ¹, Xirong Wang ¹, Beizhan Liu,² Tianwei Zheng ¹ and Zheng Wang¹

¹*Xi'an University of Science and Technology, Xi'an 710054, China*

²*SHCCIG Yubei Coal Industry Co., Ltd., Yulin 719000, China*

Correspondence should be addressed to Lihong Dong; donglh@xust.edu.cn

Received 9 February 2021; Revised 8 July 2021; Accepted 2 August 2021; Published 17 August 2021

Academic Editor: Yin Zhang

Copyright © 2021 Lihong Dong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Based on evolutionary game theory, this paper proposes a new information acquisition mechanism for intelligent mine construction, which solves the problem of incomplete information acquisition in the construction of new intelligent mining area and reduces the difficulty of information acquisition, which solves the problem of the imperfect mine information acquisition in the construction of a new smart mine regions and decreases the difficulty of a mine information acquisition. Based on the evolutionary game model, the perceptual incentive model based on group is established. The reliability of information collection is ensured by sharing and modifying the information collector. Through the analysis of the simulation results, it is found that the regional coverage model based on the cooperation in game theory and evolutionary game theory has a good effect on solving the bottleneck problem of the current intelligent mining area. This paper has an enlightening effect on the optimization of the mine information acquisition system. Through the improvement of the mine information acquisition system, the working efficiency of the information acquisition terminal can be effectively increased by 6%.

1. Introduction

Most of the mining companies in the world are traditional mining companies. They use the traditional mining information acquisition mechanism and the daily inspection information of the skilled workers as the daily operation indicators of the mine. However, the traditional production management method is often lack of the scientificity and the intelligence, which leads to the low level of information and intelligence in mining regions.

The practical experience-oriented production leads to the increased uncertainty and the increased risks in the production process. This adversely affects the life safety of miners in the long-term underground work environment. In the actual management practice, there is no unified planning and management mechanism. Many workers are unable to use information collection equipment and are reluctant to collect information in the higher risk region. This directly leads to the incomplete information acquisition [1].

Meanwhile, the measurement error and the insufficient information coverage make it impossible to conduct the sta-

ble measures in a high-risk region. This implies the danger of the safety construction of the mining region. In addition, the knowledge level of workers in the traditional mining industry is relatively lower, and the level of informatization is different, and the learning ability of the equipment is not satisfied. In the actual information acquisition process, a lot of inaccurate information will be collected.

In addition, the corporate culture of the traditionally heavy industry company leads to the vicious competition among information collectors and the poor degree of information fusion. This leads to poor integration of information between regions.

In view of the above problems, in the actual situation of the existing enterprises, this paper tries to solve the problem that the enthusiasm of the staff to collect information is not high, and this is the basis of the intelligence and controllability of information. Combining the traditional incentive model with the game theory model, an improved incentive mechanism for encouraging employee to collect and share information in high-risk region through differentiated monetary rewards within a limited budget [2, 3]. Using this

mechanism, the coverage of information collection region is maximized and the information errors caused by human factors are decreased. So, the system scheme of this paper is shown in Figure 1.

The chapters of this paper are as follows: the second part is literature review. The third part is the regional maximization incentive model. The fourth part is the evolutionary game incentive model. Finally, the fifth part is the conclusion.

2. Related Work

There are many researches on information acquisition mechanism based on evolutionary game theory.

Wang et al. (2020) used word-of-mouth crowdsourcing to design game theory algorithms to analyze the interactions between mobile contributors [4]. Based on the Stackelberg game, the behavior of the contributor is analyzed and the optimal strategy is found for the contributor. Two different modes of crowd-sourced task publishing are considered. One-time parallel publishing and multitime sequential publishing form two different market scenarios. Then, for these two cases, stage and multistage contributor game models are established, respectively. The inverse induction method is used to analyze each game, and the optimal strategy of the game is transformed into an optimization problem. Lagrange Multiplier and Karush Kuhn Tucker (KKT) methods are used to solve the optimization problem. Theoretically, the existence and uniqueness of Stackelberg equilibrium are proved.

Zeng et al. (2021) used the reputation incentive mechanism of the service cache and game theory analysis to perform edge calculations on the software-defined vehicles [5]. Contributions are measured by introducing reputation as the basis for each vehicle to provide a different quality of service. An incentive mechanism was designed using Stackelberg game modeling, and the optimal strategies of both sides of the game were analyzed by backward induction. In addition, the existence and uniqueness of the two-stage Stackelberg equilibrium is proved, and a genetic optimization algorithm is designed to quickly obtain the optimal strategy of both players in the game.

Shu et al. (2015) presented research on the extraction of integrated sensor ontology with global and local alignment [6]. He proposed to strengthen the communication mode between sensor networks in the Internet of Things and establish the semantic connection between sensor ontologies, which is essential in the communication field.

This study strengthens the communication mode between sensor networks in the Internet of Things and establishes semantic connection between sensor ontologies, which is essential in the communication field.

The research work of the initial incentive mechanism mainly focuses on the participatory perception system, some of whose research work is to improve the entertainment of the task. It is hoped that the participants can actively improve the sensory information by participating in the game. In the group-aware computing system, however, the most tasks are more urgent, and the more direct material rewards have a more positive effect on the employee's willingness to partic-

ipate. Musthag and Ganesan (2013) used monetary incentives as the main means to stimulate information collectors [7]. For the monetary incentive, the currently main researches are divided into the static monetary incentives and the dynamic monetary incentives [8, 9]. The main difference is the variability of the incentive amounts and the adjustable basis for the currently specific situation.

2.1. Static Monetary Incentive Research. In the static monetary incentives, the number of rewards is predetermined according to some criteria, and the amount is the same throughout the perception process. Reddy et al. (2010) found that the amount of payment directly affected the employee's participation and the information accuracy through adjusting the incentive amount high, medium, and low [10]. Musthag and Ganesan (2013) considered the employee's psychological factors and hides a part of the reward amount. The number of rewards can be known only after the task is completed. This method can effectively invoke the employee's expectation to the reward amount and increase the enthusiasm of the employee. However, the method did not consider the individual deviation caused by the employee's own personal factors, and the detailed setting of the variables has not been described in detail. Then, the method cannot achieve a good regression in the more specific case.

2.2. Dynamic Monetary Incentive Research. Dynamic monetary incentives set a variable budget for each task, and the budget depends on the real-time conditions. Lee and Hoh (2010) showed that the problem with fixed pricing is to determine the right amount for each task, because from an economic point of view, high prices lead to an unworkable strategy, and low prices discourage employee participation [11]. Singla and Krause (2013) used an online learning mechanism that minimizes regret [12]. Luo and Liu (2014) proposed that the task publisher can select a winner from all the participants after the task is completed or after a certain time [13]. Zhang et al. (2015) expands to a Nash bargaining game model for the multiparticipant scenarios that is perceived by the mobile group intelligence [14]. However, there is no reasonable explanation for the specific evaluation method of the game model with the multiple participants and the model evaluation deviation caused by individual differences in the model evaluation.

Adeel et al. (2014) used the virtual markets in a mobile sensing environment where the profit of information transactions is between the sender and the receiver's sales price [15]. Singla and Krause (2015) proposed an equilibrium method based on the reverse auction theory. The final price would stabilize in an interval, and the final incentive would be stabilized [16]. However, this method is limited to specific scenarios.

For a wide region of scenarios and multiparticipant balance research, there is no deep discussion. The proposed balance has certain limitations. Shaw et al. (2011) and Yan et al. (2010) explored crowdsourcing as a recruiting mechanism [17, 18]. Tham and Luo (2014) proposed a strategy that measured the quality of information and the amount of time series information and motivated participants to continue

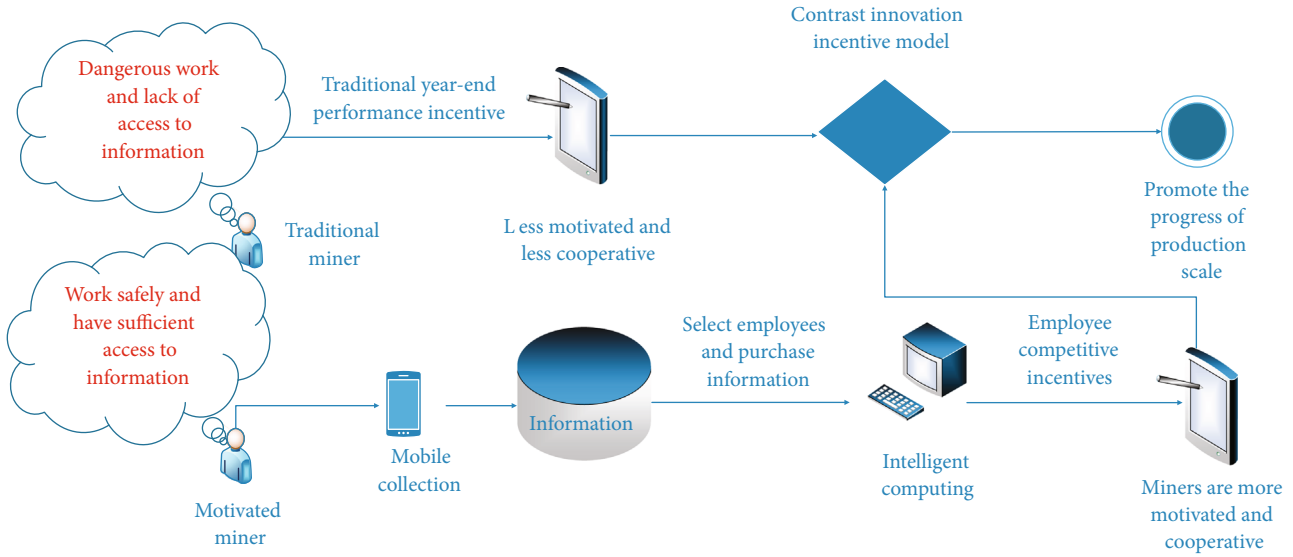


FIGURE 1: System scheme.

to participate and provide high quality information [19]. Balkanski and Hartline (2016) used Bayesian mechanism to implement the approximately optimal mechanism for achieving budget balance in a pricing incentive system [20]. Nan et al. (2014) presented a cross-space, multi-interaction-based dynamic incentive mechanism [21]. Singla and Krause (2013) presented an incentive mechanism based on the connection between reverse auctions and the multiarm bandit problem [22]. Denzis et al. (2005) used techniques from economics and psychology to address the problem of compensating people to get and use the relative information [23]. Feng et al. (2014) proposed an incentive mechanism that selects a representative subset of the participants with a constrained budget [24].

The current researches are mainly based on the theoretical models through setting simulation conditions and conducting experiments in an environment. This paper proposes the practical and feasible improvement scheme and focuses on the information acquisition incentive mechanism model. Furthermore, the game theory model is also used. In addition, the actual coal mining management is taken as the application example.

3. Region Maximization Incentive Mechanism

3.1. Reverse Auction Competing of the Evolution Game Theory. The information acquisition model collects information through the mobile employees with sensing and computing devices. The employee expenses a certain amount of personal costs in the process of participating in the information perception. Then, the knowledge gained and some equipment may be lost. In order to ensure the employee's participation in the process of information collection, it is necessary to establish a certain incentive mechanism, so that employees can get more returns and maximize the utility while paying the cost [25]. As a special game form in game theory, the auction model is often used to model the process

of information collection in group perception [26]. Using the auction model, each employee has a price tag based on the cost of the information collection process. The information collection center acts as a buyer and purchases the information collected by the employee. For the information center, it aims at information buying with the lowest price and the maximum utility. For the employee, the collected information is sold at the highest price, that is, the payments are considered to maximize the utility.

Since the auction model is a low-cost transaction, the cost of purchasing the perceived information can be reduced. But at the same time, the employee exit, the incentive cost explodes, and the information quality need to be considered due to the auction failure. Therefore, the monetary incentives are a common method of returning employees [27].

For the price selection, this paper assumes that the information center selects the lowest price in the same type of information and rejects other quotations. In information selection, this section only considers the coverage of the information, that is, whether the current information overlaps with the existing information coverage region. The region coverage maximization incentive mechanism based on reverse auction is used for the region coverage and employee participation in group perception [28]. Considering the actual scenario, the employee's price tag is combined with the employee's perceived cost.

Under the limited budget of the platform service provider, the incentive mechanism aiming at the region coverage maximization algorithm is proposed, which reduces the possibility of overlapping the employee's perceived region of platform selection. At the same time, this method keeps employees involved, increases the perceived region of the selected employees of the platform, and collects more comprehensive sensory information to improve the perceived service quality. The reverse auction is the standard requirement of the information acquisition center, and it is a revolutionary and epoch-making technology [29]. The employee

conducts the interactive real-time bidding through a dedicated network platform during the effective time. The bid at the end of the auction is the final quotation of each employee. The supply strength is given to a comprehensive evaluation, so that one or several of the most competitive employee are selected as the information collector.

3.2. Parameter Settings of the Reverse Auction Competing. In a fixed region where information collection is required, the group sensing system consists of information collection employees and information collection platforms [30]. In the initial state, N employees are randomly distributed in the specified region.

Assume that w_i represents the employee i . If each employee can perceive the same region, that is, the information amount is fixed for each employee to collect. With a radius of r , the circle represents the employee's perceived region. Then, the sensing region s_i of the employee i can be expressed as shown in Equation (1).

$$s_i = \pi r^2 (i = 1, 2, 3, \dots, N). \quad (1)$$

The employee set $W = \{w_1, w_2, w_3 \dots w_n\}$ is taken as the final information collection result. The information collection platform publishes information collection tasks based on the current production requirements and selects employee and purchases information from a centralized set of employees with a budget.

The reverse auction method is used to simulate the process of sensory information collection. The employee competes as a bidder in the reverse auction to sell the information. The service platform of the sole auctioneer selects the employee to purchase information from the optimal participating employee w_i in a certain standard. Then, the service platform notifies other unsuccessful employees to win or lose the result of this round of auction, and the winner obtains the price. As the income, the participating employee judges the next round state. If the employee exits, the service platform informs the exiting employee of the maximum price of the auction, and the exiting employee judges whether the next round of auction returns.

In the process of information auction, each employee has an initial bid price a_i and the cost c_i for acquiring information according to the individual situation.

This paper defines the employee participation cost as the sum of completing the information sensing and sending the sensory information. Because all employees have the same perceived region, the difference among the cost of participants is the difference in the transmission cost. If the communication fee for the employee to send the information is borne by the platform, the employee participation cost is mainly the power consumed. Referring to the path loss index in different environments, the propagation loss of the channel in the urban environment is proportional to the fourth power of the distance. The employee participation cost c_i is defined by Equation (2).

$$c_i = \alpha d_i^4 (i = 1, \dots, N), \quad (2)$$

where α represents the price per unit distance and d_i represents the distance between the employee and the base station.

The information center publishes the information requirement with a low price. This may result in overlapping of the perceived region, that is, the information reported by the adjacent employees is the same, and the service platform wastes funds on the budget. On the other hand, the partial region information cannot be obtained, which reduces the coverage of the perceived information.

The regional coverage maximization algorithm can solve the above problem [31]. The goal is to make the selected employee's perception regions not overlap each other if the budget L is fixed.

If the employee w_1 be selected in the employee set W , define the number of employees who are aware of the scope of all employees in the participating employee u_i . The selected employee set w_1 as the overlapping region of the participating employees u_i . The employee-aware region overlap degree cdd_i is given by Equation (3).

$$cdd_i = \begin{cases} cdd_i + 1, & \text{If } d(u_i, u_j) < 2r, \\ 0, & \text{Otherwise,} \end{cases} \quad (3)$$

where $d(u_i, u_j)$ represents the distance between two employee centers, $u_i \subseteq W, u_j \subseteq W_1$.

The specific process is as follows:

- (1) Let $W = \{u_1, u_2, \dots, u_N\}$, $W_1 = \emptyset$, loop counter $m = 0$
- (2) The employee with the lowest price from the participating employees and W is selected as the winning employee in the selected employee set w_1
- (3) Select an employee with the employee-aware region overlap degree $cdd_i = m$ and the lowest price in the selected employee set w_1 from the remaining set $w \setminus w_1$ into the selected employee set w_1 , and subtract the employee from the budget L price
- (4) Determine whether the budget L is exhausted. If there is still a budget, go to step (5); if the budget is exhausted, output the selected employee set w_1 and end the algorithm
- (5) Judging the set $w \setminus w_1$, there is still employee overlap with the perceived region of the employee in w_1 , $cd d_i = m$; if there is still, then carry out step (3) and step (4). If not, $m = m + 1$, and return to step (3), until the budget L is exhausted or the participating employees W are all selected. The selected employee set w_1 is obtained and the algorithm ends

When the platform selects the employee to end, all the employees in the winning employee set w_1 sell the sensory information set to the platform and obtain the price as the revenue. The failed employee is not selected by the platform in this round of auction, and the profit is zero. Participating in the total revenue earned by the employee statistics participating in the auction, when the total revenue is lower than

the employee participation cost, the employee chooses to exit the next round of auction.

Since the participation cost c_i of the employee in the scenario is directly related to the distance d_i between the employee and the base station, the profit rate G_i^r is defined. The profit rate G_i^r of the participating employee u_i at the r round auction is given by Equation (4).

$$G_i^r = \frac{e_i^r}{p_i^r c_i}, \quad (4)$$

where e_i^r identifies the total revenue obtained by the employee u_i from the first round to the r round auction. If the employee u_i wins in the r round, then $e_i^r = e_i^{r-1} + b_1^r$, where b_1^r indicates the bid of the employee u_i in the r th round. If the employee u_i fails in the r th round, $e_i^r = e_i^{r-1}$. Indicating the total number of participations of the employee u_i from the first round to the r round auction, c_i represents the participation cost of the employee. $p_i^r c_i$ represents the total cost value of the employee u_i from the first round to the r round. For each round of auction, each employee participates in the employee to calculate the rate of return. If the employee's G_i^r is higher than a certain threshold, it means that the employee is still participating in the next round of auction, and the bid price is appropriately adjusted. If it is below a certain threshold, then the next round of this employee exits.

For the exiting employee u_k , the platform informs the r round maximum transaction price φ_r . The employee calculates its budget period yield EG_k^{r+1} in the $r+1$ th round. The EG_k^{r+1} is given by Equation (5).

$$EG_k^{r+1} = \frac{e_k^r + \varphi_r}{(p_k^r + 1)c_k}. \quad (5)$$

The incentive mechanism is mainly composed of platform selection and employee judgment. The platform selection criterion based on the regional coverage maximization algorithm guarantees the regional coverage rate, and the employee judgment guarantees the employee participation in group perception. The employee first calculates the participation cost based on the distance of the base station, then sets the initial price in the auction according to the participation cost and participates in the auction. According to the regional coverage, the service platform acts as the auctioneer in the case of a fixed budget. The algorithm selects the employee to purchase the sensory information. Participating employees judge whether to continue to participate in the auction according to the rate of return. For the exiting employee, the platform will inform the maximum price in the second round of auction, and the exiting employee calculates the expected rate of return to determine whether to return to the auction.

3.3. Simulation Result Analysis. In a fixed region where information collection is required, the group sensing system consists of information collection employees and information

collection platforms. In the initial state, N employees are randomly distributed in the specified region, and w_i is used to represent employee i . If each employee can perceive the same region, the amount of information that each employee can measure is certain. With a radius of r , the circle represents the employee's perceived region. Then, the sensing region of the employee i can be given by Equation (6).

$$s_i = \pi r^2 (i = 1, 2, 3, \dots, N). \quad (6)$$

For example, the number of participating employees $N = 200$, budget $L = 4000$, and employee perceived radius $r = 16$. In the iterative process, the participating employees judge whether to participate in the next round of auction and adjust the price according to the rate of return after each auction.

As a result, the uncertainty of the employee's decision for the next round leads to the volatility and system of the participating employees' quotations. By exiting the employee to calculate the expected rate of return to determine whether to return, it is possible to ensure that the number of participants in the system is maximized. Therefore, the average incentive price of the selected employee converges after a certain number of auction iterations, and the employee price is finally stable, that is, the currently selected region. The relationship between iteration number and price relationship is shown in Figure 2.

As the platform chooses as much as possible, the sensing region does not overlap. The employee distribution diagram is shown in Figure 3.

As the perceived radius r increases, the degree of overlap of the employee's perceived region is higher. In the case of L fixed, the criterion for the platform to select employees is to select unselected employees with the selected employee's perceived region. If the perceived radius r of the employee is small, the employee perception region hardly overlaps or the degree of overlap is low. Since the employee participation cost is related to the distance from the base station, the employee price around the base station is low, and the service platform selects the employee around the base station to cause the purchase under a fixed budget. If the perceived radius r of the employee is large, the employee's perceived region overlaps a lot. In order to avoid overlapping of the selected employee's perceived region, the selected employees are dispersed in the target region, so the number of employees purchased under the same budget is small.

When the employee's perceived radius is the same, the greater the L , the larger the number of employees that can be purchased under different budgets L . The relationship between budget and selected employee is shown in Figure 4.

Analysis shows that the larger the budget of the service platform, the more employees choose to purchase. The service platform can purchase the perceptual information of all participating employees when the budget is large enough. Therefore, regardless of the employee radius r , the number of selected employees is equal to the number of participating employees when the budget is large enough. When the

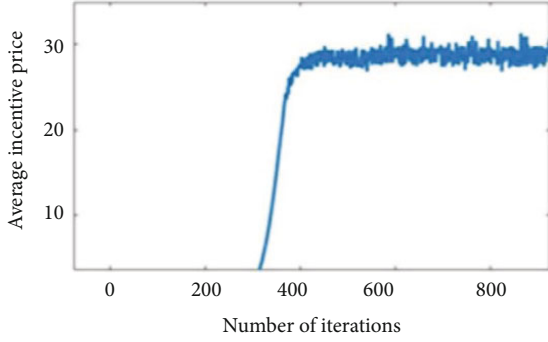


FIGURE 2: The relationship between iteration number and price relationship.

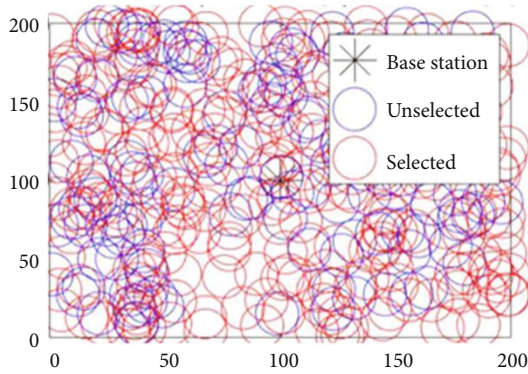


FIGURE 3: The diagram of employee distribution.

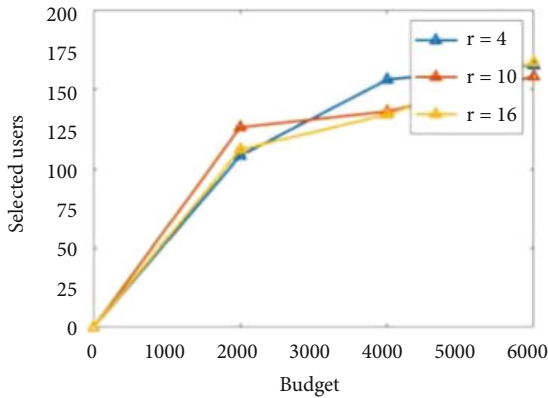


FIGURE 4: The relationship between budget and selected employee.

perceived radius r is small, the employee's perceived region hardly overlaps or the degree of overlap is low. Since the price of the employee around the base station is low, the service platform starts to select from the employees around the base station. As a result, the number of employees purchased under the fixed budget is large. The relationship between employee perception radius and the number of employees is shown in Figure 5.

4. Evolutionary Game Incentive Model

4.1. *Game Strategy and Revenue Matrix.* In the perception system, when only participant B selects the shared informa-

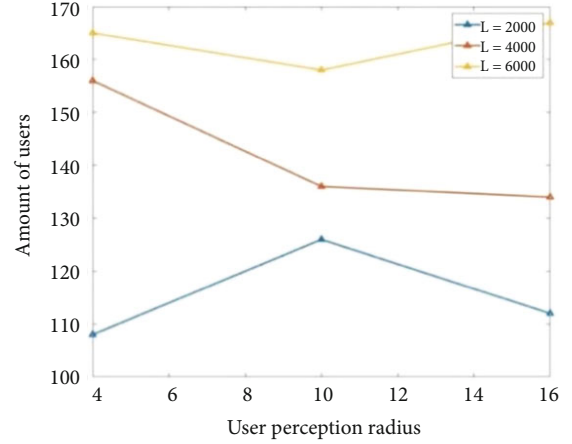


FIGURE 5: The relationship between employee perception radius and the number of employees.

tion, participant A obtains the perceptual information of participant B. Therefore, participant A obtains more accurate perceptual information through information processing and receives more rewards from the platform. For participant B, sharing information does not make it more rewarding. B's strategy in the next round of evolutionary games is uncertain, which may affect the enthusiasm of B cooperation [32]. In order to encourage participants to cooperate in the game and promote information exchange, it is necessary to set reward and punishment measures in the game model. This paper introduces the cooperation rate as an incentive penalty factor to promote the cooperation of participants [33]. According to the cooperation rate, the participant's game cost can be dynamically adjusted. The participant with high cooperation rate will reduce the game cost, which is a reward for the cooperation of the participants. On the contrary, participants with low cooperation rates will increase the cost of the game, which is also to increase the cooperation rate through punishment. Assuming that the evolutionary game of w round has been experienced in the recent time period, the cooperation rate of each round of participants can be expressed as $\{x_1, x_2, x_3, \dots, x_w\}$, then the cooperation rate of the most recent period is given by Equation (7).

$$z = \frac{1}{w} \sum_{k=1}^{\infty} x_k y(k), \quad (7)$$

where $y(k) \in [0, 1]$ is the attenuation function, so the adjustment coefficient β of the game cost t can be expressed by Equation (8).

$$\beta = 1 + \eta - z, \quad (8)$$

where $0 < \eta < 1$. When the cooperation rate is 1, the game cost $\beta t = \eta t$; when the cooperation rate is 0, the game cost is $\beta t = \eta t$ currently. In the actual scenario, the company's budget is not unlimited. The participant's return is the

amount paid by the information platform to the participant, that is, the total expenditure of the service platform is the total return sum of all participants. Define α to adjust the income $s + \Delta s$ to control the amount of payment. The income adjustment coefficient α can be expressed by Equation (9).

$$\alpha = 1 + \frac{u_{th} - u_j^{\text{total}}}{u_{th}}, \quad (9)$$

where u_{th} is the platform budget, u_j^{total} is the total expenditure of the platform after the j -round game.

The main idea of the evolutionary game is “survival of the fittest,” that is, it should encourage behaviors that promote the evolution of the system and impose penalties on behaviors that hinder the evolution of the system. Define the employee’s revenue as the fitness in the evolution, continuously generate high-yield employees, and eliminate low-yield employees. After a round of evolutionary game, each participant calculates its income. If the return is higher than the average of employee’s return, the system will increase its number of games in the next round of evolution. On the contrary, if the employee’s return is lower than the average of employee’s return, the number of games will be reduced in the next round of games. Let $f_{i,j+1}$ denote the number of players in the $j + 1$ round, then the equation of $f_{i,j+1}$ is given by Equation (10).

$$f_{i,j+1} = f_{i,j} + f_{i,j} \frac{u_{i,j} - \bar{u}_j}{u_{i,j}}, \quad (10)$$

where $f_{i,j}$ represents the number of games in the j th round of the participants, $u_{i,j}$ represents the participant’s return in j round, $\bar{u}_j = u_j^m / m_j$ is the j th round average revenue of the employee, and m_j represents the number of participants in the j th round. In order to survive, participants in different strategies constantly adapt to the environment during the evolutionary game, and each participant updates their strategy with a certain probability. Copy rules, also known as proportional imitation rules, mean that when a round of evolutionary games ends, participants randomly imitate another participant with a certain probability. The participant update policy rule is defined by Equation (11).

$$P(\pi_A \longrightarrow \pi_B) = \begin{cases} \delta, & u_n > u_A, \\ 0, & u_n \leq u_A, \end{cases} \quad (11)$$

where π_A and π_B represent the strategies of participants A and B, u_A and u_B represent the returns of participants A and B, and δ is the probability coefficient. The above formula indicates that when the benefit of B is greater than the return of A, the probability of adopting B’s strategy is an update of its own strategy.

Assuming that the total number of participants in the system is m , each participant has two game strategy choices, shared or not shared. u_{th} represents the budget set by the service platform. The median income $s + \Delta s$ and the game cost t

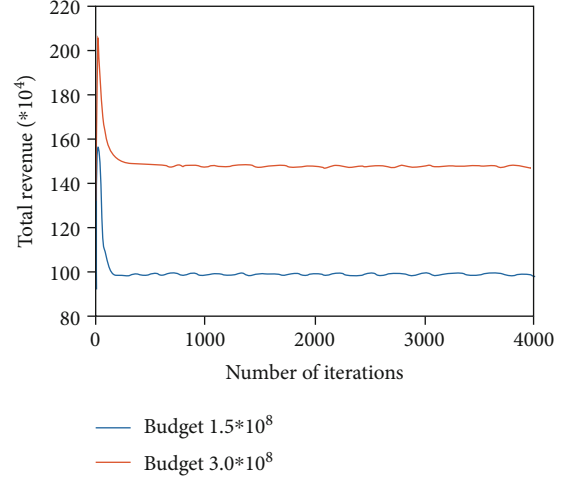


FIGURE 6: System stability.

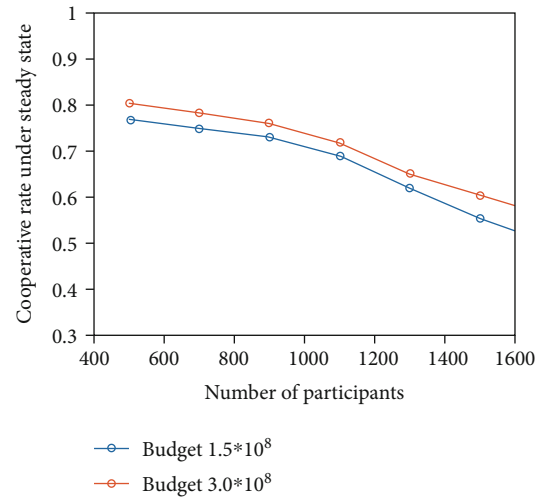


FIGURE 7: Relationship between cooperation rate and number of participants in steady state of the system.

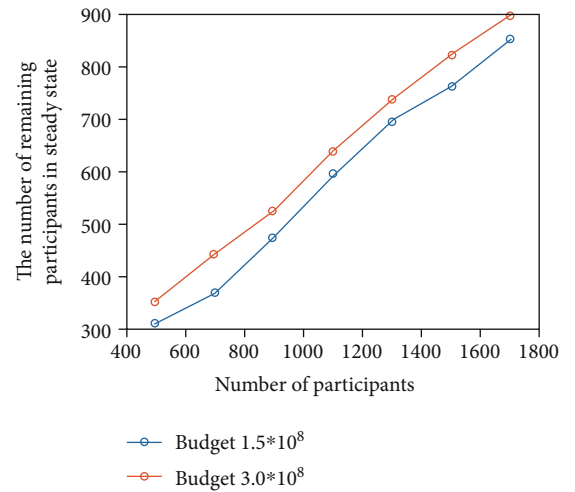


FIGURE 8: Relationship between remaining participants and number of participants in steady state.

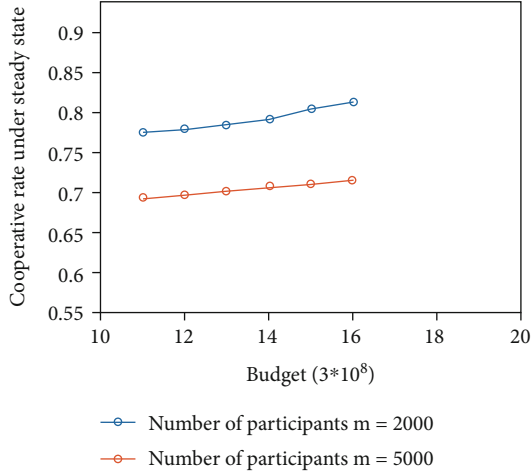


FIGURE 9: Relationship between cooperation rate and the budget.

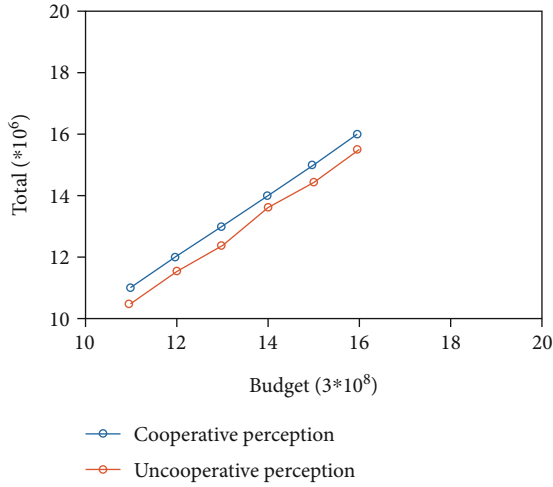


FIGURE 10: The income changes after the introduction of the cooperation mechanism.

of the initial return matrix are all positive integers. The initial cooperation rate of each participant is 0.5, and the initial number of games is f . Each participant performs f -games with other $m-1$ participants, respectively. This process is called a round of evolution. After a round of game, calculate the income of each participant, so that the income of i participants in the j th game is s_i^j , the cost is t_i^j , then the total return is $u_{i,j}^{\text{total}} = s_i^j - t_i^j$. The total employee income is $u_j^{\text{total}} = \sum_{i=1}^m u_{i,j}^{\text{total}}$, and the average return is $\bar{u}_j = (1/m) \sum_{i=1}^m u_{i,j}^{\text{total}}$. Then, adjust α and β , update the return matrix $s + \Delta s$ and t , and calculate the number of times each participant plays in the next round. If it is 0, exit the game until the system reaches stability.

4.2. Simulation Result Analysis. This paper first analyzes the stability of the system and simulates the iterative curve of the total revenue of the system steady state under the $u_{th} = 3.0 \times 10^8$ and $u_{th} = 1.5 \times 10^8$, respectively. The algorithm uses α to adjust the income to achieve control over the budget

amount. As shown in the simulation results, the total revenue of the employee does not exceed the total budget of the platform. Adjusting the parameter $\alpha > 1$, the next round of $s + \Delta s \rightarrow \alpha(s + \Delta s)$ will increase with the number of iterations. In the early stage of the game, the employee's income increases. When $\alpha < 1$, the employee's income can be reduced.

By constantly adjusting the parameters, the employee can get the maximum benefit within the budget and finally stabilize at a fixed value. At the moment, the system is stable, and the employee income can be recorded as a stable income. The system stability is shown in Figure 6.

Under the budget of $u_{th} = 3.0 \times 10^8$ and $u_{th} = 1.5 \times 10^8$, the relationship between the cooperation rate and the number of participants in the steady state is discussed in this paper. When the number of participants in the system increases, the budget of the platform is fixed. The competition among participants increases with the increase of people, resulting in a significant decline in the cooperation rate.

Relationship between cooperation rate and number of participants in steady state of the system is shown in Figure 7.

In the case of a fixed number of participants, the increase in budget promotes cooperation among participants, so the cooperation rate increases with the budget when the system reaches stability. By introducing a reward factor, as the number of participants increases, the number of reputation participants in steady state also increases, which is indicating that the system is efficient. By encouraging more participants to join the information sharing, employee engagement can effectively increase.

Under the budget of $u_{th} = 3.0 \times 10^8$ and $u_{th} = 1.5 \times 10^8$, the relationship between remaining participants and number of participants in steady state of the system is shown in Figure 8.

Finally, in the condition of the system reaching a steady state, the relationship between the cooperation rate and the budget is simulated. It can be seen from the simulation results that the increase of the budget promotes the cooperation between the participants, so the cooperation rate becomes larger as the budget increases. The relationship between cooperation rate and the budget in system steady state is shown in Figure 9.

In the case of a fixed budget, the greater the number of participants, the more intense the competition between them. Therefore, the cooperation rate under steady state is reduced. In the process of evolutionary game, the fitness is the employee's benefit. The participants in the group continuously improve their strategies and learn from the high-yield partners. Therefore, when the system reaches a stable state, the number of employees who choose to cooperate will increase a lot. The total employee income is maximized within the budget, which increases the total revenue of the employee compared to the perception of noncooperation.

When the number of participants is $m = 2000$ and $m = 5000$, the income change after the introduction of the cooperation mechanism is shown in Figure 10.

Considering the limited number of employees and possible experimental deviations, this paper conducted a relevant robustness test at the end of the experiment, trying to pass multiple and diverse parameters. This model simulation

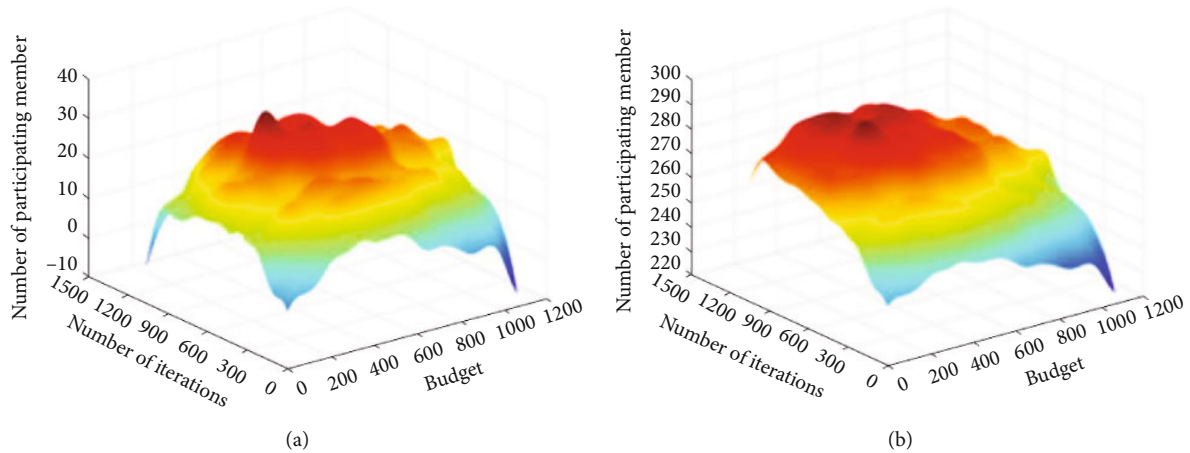


FIGURE 11: Multivariate simulation results.

eliminates the deviation of experimental results caused by a single parameter change. Therefore, this paper will run a simulation in a linearly increasing of the employees' number, iterations, and the budget.

By changing the employees' and iterations' number, increasing the budget accordingly, the effectiveness of model simulation can be effectively improved. Therefore, based on the simulation effort model proposed in this paper, it can be found that budget control and employee number control are both important parameters for completing game equilibrium. By effectively improving the incentive budget and the number of participating employees in actual production, the incentive model is guaranteed to achieve maximum utility. Multivariate simulation results are shown in Figure 11.

Based on the above analysis results, the proposed algorithm effectively improves the incentive mode. Compared with traditional year-end result incentives, competitive incentives enable first-line miners to achieve good competition and cooperation in their daily work and increase the enthusiasm of employees. Therefore, the incentive algorithm proposed in this paper is feasible and consistent with the actual situation, which can better promote the work efficiency of employees and the reform of corporate strategic management.

5. Conclusions

The experimental results show that after the introduction of incentive mechanism, the budget required by the enterprise is 6% lower than that of the traditional enterprise, the cooperative relationship between employees is more stable, and the enthusiasm of employees will be further improved. Therefore, multiple conclusions can be drawn.

- (1) Company can maximize the scope of information collection by effectively establishing incentives. Under a limited budget, aiming at maximizing regional coverage, the reverse auction principle is used to motivate information collectors collecting information from remote regions

- (2) Through the evolutionary game model, this paper introduces the concept of reward factor by simulating the cooperative game behavior and strategy update behavior of workers in the collection process. This method encourages all information collectors sharing information by providing additional subsidies to information sharers and solves problem of inaccurate information collection

- (3) Considering that the algorithm proposed in this paper is costly in practice, this paper carries out further feasibility tests. By selecting the information of the same type of mining enterprises, it is found that the management reform can effectively promote the progress of production scale. Therefore, it can be inferred that the management reform proposed in this paper is reasonable in practice and effective at solving the current production problems caused by improper management

The future development of information acquisition incentive mechanism in intelligent mine construction mainly includes two aspects: the first is the theoretical system, which further innovates the existing theories and expands the scope of application of theories. Second, the application of the incentive mechanism, to further improve the technology needed in the incentive mechanism, the incentive mechanism will be better applied in practice, become an important link in the construction of intelligent mines.

Data Availability

The data used to support the findings of this study have not been made available because data privacy.

Conflicts of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This paper is supported by the National Natural Science Fund grant no. 51804249.

References

- [1] Y. S. Dohare, T. Maity, P. S. Das, and P. S. Paul, "Wireless communication and environment monitoring in underground coal mines – review," *IETE Technical Review*, vol. 32, no. 2, pp. 140–150, 2015.
- [2] M. Askarizadeh, B. Tork Ladani, and M. H. Manshaei, "An evolutionary game model for analysis of rumor propagation and control in social networks," *Physica A: Statistical Mechanics and its Applications*, vol. 523, pp. 21–39, 2019.
- [3] M. Chica, R. Chiong, M. T. P. Adam, and T. Teubner, "An evolutionary game model with punishment and protection to promote trust in the sharing economy," *Scientific Reports*, vol. 9, no. 1, p. 19789, 2019.
- [4] R. Wang, F. Zeng, L. Yao, and J. Wu, "Game-theoretic algorithm designs and analysis for interactions among contributors in mobile crowdsourcing with word of mouth," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8271–8286, 2020.
- [5] F. Zeng, Y. Chen, L. Yao, and J. Wu, "A novel reputation incentive mechanism and game theory analysis for service caching in software-defined vehicle edge computing," *Peer-to-Peer Networking and Applications*, vol. 14, no. 2, pp. 467–481, 2020.
- [6] S. Fu, H. Wen, J. Wu, and B. Wu, "Energy-efficient precoded coordinated multi-point transmission with pricing power game mechanism," *IEEE Systems Journal*, vol. 11, no. 2, pp. 578–587, 2015.
- [7] M. Musthag and D. Ganesan, "Labor dynamics in a mobile micro-task market," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 641–650, Toronto, Ontario, Canada, April 2013.
- [8] R. Becker and D. Glauser, "Are prepaid monetary incentives sufficient for reducing panel attrition and optimizing the response rate? An experiment in the context of a multi-wave panel with a sequential mixed-mode design," *Bulletin of Sociological Methodology/Bulletin de Méthodologie Sociologique*, vol. 139, no. 1, pp. 74–95, 2018.
- [9] D. Bird and A. Frug, "Dynamic non-monetary incentives," *American Economic Journal: Microeconomics*, vol. 11, no. 4, pp. 111–150, 2019.
- [10] S. Reddy, D. Estrin, M. Hansen, and M. Srivastava, "Examining micro-payments for participatory sensing data collections," in *Proceedings of the 12th ACM international conference on Ubiquitous computing*, pp. 33–36, Melbourne, Australia, September 2010.
- [11] J. Lee and B. Hoh, "Dynamic pricing incentive for participatory sensing," *Pervasive and Mobile Computing*, vol. 6, no. 6, pp. 693–708, 2010.
- [12] A. Singla and A. Krause, "Truthful incentives in crowdsourcing tasks using regret minimization mechanisms," in *Proceedings of the 22nd international conference on World Wide Web - WWW '13*, pp. 1167–1178, Rio de Janeiro, Brazil, May 2013.
- [13] H. Luo and S. Liu, "Effect of situational leadership and employee readiness match on organizational citizenship behavior in China," *Social Behavior and Personality: An International Journal*, vol. 42, no. 10, pp. 1725–1732, 2014.
- [14] H. Zhang, C. Jiang, N. C. Beaulieu, X. Chu, X. Wang, and T. Q. S. Quek, "Resource allocation for cognitive small cell networks: a cooperative bargaining game theoretic approach," *IEEE Transactions on Wireless Communications*, vol. 14, no. 6, pp. 3481–3493, 2015.
- [15] U. Adeel, S. Yang, and J. A. McCann, "Self-optimizing citizen-centric mobile urban sensing systems," in *11th International Conference on Autonomic Computing*, pp. 161–167, Philadelphia, Pennsylvania, 2014.
- [16] B. Guo, H. Chen, Z. Yu, X. Xie, S. Huangfu, and D. Zhang, "FlierMeet: a mobile crowdsensing system for cross-space public information reposting, tagging, and sharing," *IEEE Transactions on Mobile Computing*, vol. 14, no. 10, pp. 2020–2033, 2015.
- [17] A. D. Shaw, J. J. Horton, and D. L. Chen, "Designing incentives for inexpert human raters," in *Proceedings of the ACM 2011 conference on Computer supported cooperative work - CSCW '11*, pp. 275–284, Hangzhou, China, 2011.
- [18] T. X. Yan, V. Kumar, and D. Ganesan, "Crowdsearch: exploiting crowds for accurate real-time image search on mobile phones," in *Proceedings of the 8th international conference on Mobile systems, applications, and services - MobiSys '10*, pp. 77–90, San Francisco, CA, USA, 2010.
- [19] C. K. Tham and T. Luo, "Quality of contributed service and market equilibrium for participatory sensing," *IEEE Transactions on Mobile Computing*, vol. 14, no. 4, pp. 829–842, 2014.
- [20] E. Balkanski and J. D. Hartline, "Bayesian budget feasibility with posted pricing," in *Proceedings of the 25th International Conference on World Wide Web*, pp. 189–203, Montreal, Quebec, Canada, April 2016.
- [21] W. Nan, B. Guo, S. Huangfu, Z. Yu, H. Chen, and X. Zhou, "A cross-space multi-interaction-based dynamic incentive mechanism for mobile crowd sensing," in *2014 IEEE 11th Intl Conf on Ubiquitous Intelligence and Computing*, Washington, USA, 2014.
- [22] Y. Zhang, L. Song, W. Saad, Z. Dawy, and Z. Han, "Contract-based incentive mechanisms for device-to-device communications in cellular networks," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 10, pp. 2144–2155, 2015.
- [23] G. Danezis, S. Lewis, and R. Anderson, "How much is location privacy worth?," in *Proceedings of the Fourth Workshop on the Economics of Information Security*, Pittsburgh, PA, USA, 2005.
- [24] Z. Feng, Y. Zhu, Q. Zhang, L. M. Ni, and A. V. Vasilakos, "TRAC: truthful auction for location-aware collaborative sensing in mobile crowdsourcing," in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, pp. 1231–1239, Toronto, Ontario, Canada, April 2014.
- [25] Y. Wang, Z. Cai, G. Yin, Y. Gao, X. Tong, and G. Wu, "An incentive mechanism with privacy protection in mobile crowdsourcing systems," *Computer Networks*, vol. 102, pp. 157–171, 2016.
- [26] M. H. Cintuglu, H. Martin, and O. A. Mohammed, "Real-time implementation of multiagent-based game theory reverse auction model for microgrid market operation," *IEEE Transactions on Smart Grid*, vol. 6, no. 2, pp. 1064–1072, 2015.
- [27] L. Cassar, "Job mission as a substitute for monetary incentives: benefits and limits," *Management Science*, vol. 65, no. 2, pp. 896–912, 2019.
- [28] H. Zhou, X. Chen, S. He, J. Chen, and J. Wu, "DRAIM: a novel delay-constraint and reverse auction-based incentive mechanism

- for WiFi offloading,” *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 4, pp. 711–722, 2020.
- [29] M. Huang, X. Qian, S. C. Fang, and X. Wang, “Winner determination for risk aversion buyers in multi-attribute reverse auction,” *Omega*, vol. 59, pp. 184–200, 2016.
- [30] B. Tian, Y. Yuan, H. Zhou, and Z. Yang, “Pavement management utilizing mobile crowd sensing,” *Advances in Civil Engineering*, vol. 2020, Article ID 4192602, 16 pages, 2020.
- [31] D. Tong, “Regional coverage maximization: a new model to account implicitly for complementary coverage,” *Geographical Analysis*, vol. 44, no. 1, pp. 1–14, 2012.
- [32] X. Chen and M. Perc, “Optimal distribution of incentives for public cooperation in heterogeneous interaction environments,” *Frontiers in Behavioral Neuroscience*, vol. 8, p. 248, 2014.
- [33] R. Terpend and D. R. Krause, “Competition or cooperation? Promoting supplier performance with incentives under varying conditions of dependence,” *Journal of Supply Chain Management*, vol. 51, no. 4, pp. 29–53, 2015.

Research Article

Forensic Analysis of Social Networking Applications on an Android Smartphone

Anoshia Menahil,¹ Waseem Iqbal¹,^{ORCID} Mohsin Iftikhar,² Waleed Bin Shahid,¹ Khwaja Mansoor,³ and Saddaf Rubab¹

¹Department of Information Security, National University of Sciences and Technology (NUST), Islamabad 44000, Pakistan

²School of Computing and Mathematics, Charles Sturt University, Australia

³Faculty of Computing and AI, Department of Cyber Security, Air University, Islamabad 44000, Pakistan

Correspondence should be addressed to Waseem Iqbal; waseem.iqbal@mcs.edu.pk

Received 20 February 2021; Revised 15 March 2021; Accepted 1 July 2021; Published 23 July 2021

Academic Editor: Ernestina Cianca

Copyright © 2021 Anoshia Menahil et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Smartphone users spend a substantial amount of time in browsing, emailing, and messaging through different social networking apps. The use of social networking apps on smartphones has become a dominating part of daily lives. This momentous usage has also resulted in a huge spike in cybercrimes such as social harassing, abusive messages, vicious threats, broadcasting of suicidal actions, and live coverage of violent attacks. Many of such crimes are carried out through social networking apps; therefore, the forensic analysis of allegedly involved digital devices in crime scenes and social apps installed on them can be helpful in resolving criminal investigations. This research is aimed at performing forensic investigation of five social networking apps, i.e., Instagram, LINE, Whisper, WeChat, and Wickr on Android smart phones. The essential motivation behind the examination and tests is to find whether the data resides within the internal storage of the device or not after using these social networking apps. Data extraction and analysis are carried out using three tools, i.e., Magnet AXIOM, XRY, and Autopsy. From the results of these experiments, a considerable amount of essential data was successfully extracted from the examined smartphone. This useful data can easily be recovered by forensic analysts for future examination of any crime situation. Finally, we analyzed the tools on the basis of their ability to extract digital evidences from the device and their performance are examined with respect to NIST standards.

1. Introduction

Smartphones have gone through a progressive advancement over the last two decades. From being a prized possession, they have become an outright need as they are used for a lot more than just calls and messages. Smartphones enable users to enjoy a comfortable internet experience by browsing, emailing, and staying connected through different social media apps. The use of social media apps has witnessed a momentous surge, and they are being used by people of all age groups, by businesses, academia, media, hacktivists, law enforcement agencies (LEAs), and even terrorist organizations for a wide variety of purposes [1]. During the last 12 months, the use of social media has continuously increased by 1 million every day [2]. According to the Digital 2020

Global Overview, the usage of social media has increased by 12 percent from 2019 and 99 percent of that surge was due to smartphone apps [2].

For most teenagers, the most popular usage of mobile phone is socializing. These online interactions are mostly productive; however, the online interactions have resulted in increased number of bullying, threatening, and humiliating others as well [3]. It is estimated that 73% of students feel that they have been harassed in their lives and 44% say they have been harassed during the last 30 days [4]. Easy access to social media platforms opens a door for a new type of bullying.

Social networking apps (SNAs) allow users to create a profile, upload personal information such as pictures, videos, and location, and share that information through private messaging or public posts. This phenomenon gives criminals

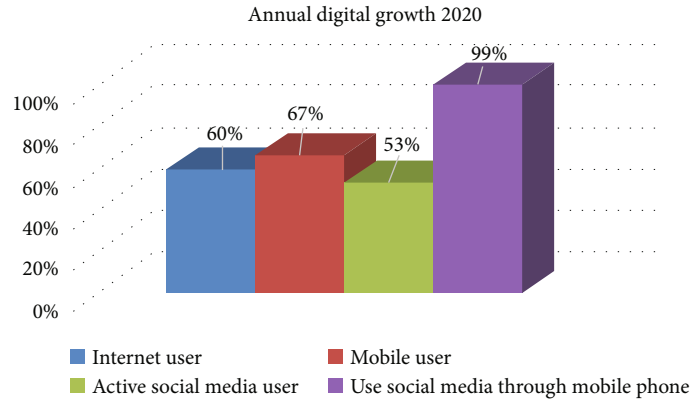


FIGURE 1: Statistics of the usage of the Internet and social media.

an open opportunity to influence the user's personal information thus giving rise to cybercrimes via SNAs [5]. These apps can also be exploited for cyber bullying, stalking, sexual harassment, and insults [6, 7]. Since all information about user activities is stored within the phone's internal memory, these smartphones have become an important source of evidence and artifacts during crime investigations linked to these SNAs. These gathered artifacts enable users and investigators to find out the PII (personally identifiable information) [8] that is stored in the device and is useful to be presented in the court of law.

The increase in cybercrimes using smartphones and SNAs [9] has created an opportunity to use forensic tools and techniques to investigate these criminal activities. The forensic analyst requires an up-to-date understanding of what kind of artifacts can possibly be recovered so that they can be presented in the court of law. Moreover, the increased demand for more and more advanced smartphones has created fierce competition among equipment manufacturers. New smartphones are continuously launched in the market that has resulted in a frequent change of mobile phones by users based on factors such as better OS, file structure, data storage, user experience, and many more [10]. Therefore, forensic examiners are struggling to keep up with new procedures and tools.

Many studies have been conducted on Facebook and WhatsApp forensic and other well-renowned apps as mentioned in literature, so there is a need for forensics analysis of other social media apps (Instagram, LINE, WeChat, Whisper, and Wickr) which are getting popular these days. With expanded use of smartphones for social networking as highlighted in Figure 1, a lot of knowledge on forensic artifacts depicting user behaviour is generated through these social networking apps [2] which are stored within the phones. Several situations involving significant accidents might occur that may end up in the court. Forensic investigators can use those activities as evidence in investigating such incidents.

Keeping this scenario in mind, we have focused on inspecting the phone forensically and looking for the artifacts generated and stored in different locations in the phone, via the forensic tools. Such artifact findings are capable of tying the perpetrator to the incident; thus, forensic examination

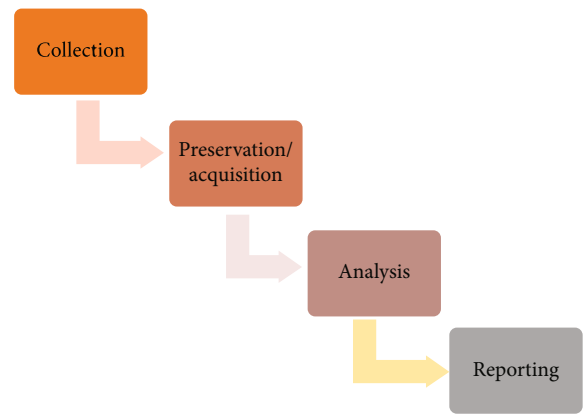


FIGURE 2: NIST forensic analysis process.

of these apps can provide a ready reckoner to the digital forensic investigator of Android phone analysis.

Thus, focusing on this requirement, our research aims at performing the following actions:

- (i) Identification, extraction, and analysis of the artifacts recovered from the popular social networking apps on an Android smartphone in a way that can be presented in the court of law
- (ii) Analysis of each app using three different tools, i.e., Magnet AXIOM, Autopsy, and XRY
- (iii) Presentation of the analysis of the tools to showcase their ability to extract digital evidences from the device and their performance is examined with respect to NIST standards

We have used three forensic tools to perform experiments on five popular social media apps including Instagram [11], LINE [12], Whisper [13], WeChat [14], and Wickr [15]. Data from these apps is acquired and analyzed during three stages (before data deletion, after data deletion, and after app uninstall) using Magnet AXIOM [16], Autopsy [17], and XRY [18]. Furthermore, we have also categorized the tools on the basis of total artifacts recovered and NIST standards on smartphone extraction tools [19]. We have also devised some additional parameters that can be used

TABLE 1: Literature study of forensic investigation on mobile applications.

App	Platform used	Artifacts	Remarks
Snapchat [34]	iOS and Android	iOS images and videos are fully detected. Android images and videos not fully detected	No correlation found between _le on XML record and recovered video.
Wickr [35]	Android	No artifacts found	After analyzing app on rooted and nonrooted devices, Wickr was proved as an antiforensic app
WhatsApp, Viber [36]	Froyover 2.2, GingerBread v2.3.x, and IceCreamSandwich v4.0.x	WhatsApp: message, timestamps, and names of files sent and received. Viber: no file found through the analyzer	Research does not focus on the artifact discovery when data was deleted
WhatsApp [37]	Android	Phone numbers, messages, media files, contact cards, and profile pictures	They successfully developed a tool to acquire data from phone
20 applications [38]	Android v4.4.2, iOS 7.1.2	Snapchat, Tinder, Wickr, and BBM: data not recovered	Main focus was on the network traffic analysis
ChatSecure [39]	Android	Recovered messages, video, and audio that have been shared. No deleted data recovered	Artifact extraction done only from the volatile memory. An algorithm was designed to decrypt the encrypted chat secure database
Kik Messenger [40]	Android v4.4,5.0	Chat messages (send/received), contacts, and media that have been exchanged are recovered	Device backup was created through the logical acquisition
Instagram [41]	iPhone 6s (iOS 10.3.0)	User information, activity history, and application settings recovered from the iPhone backup file	Backup created through logical acquisition. Only the file folder was considered for the artifacts
WhatsApp [42]	Android v5.0.1	Text messages, images, video, and documents recovered	Calculate and compare the Belkasoft Evidence (trial version) and WhatsApp Key/DB Extractor tools on the basis of artifacts recovered from WhatsApp
Telegram [43–45]	Android, Windows	Android gives messages and cache info after exploring databases. App package was not recovered in Windows phone	Analysis was done on 5 scenarios that were divided into 17 smaller scenarios but these scenarios were not discussed properly. Reference [44] did not analyze the whole _le system (part analysis has been done). Reference [45] claimed that commercial tools like Internet Evidence Finder, UFED Physical Analyzer, and Oxygen Forensic Analysis did not have the capability to acquire artifacts from Telegram Messenger for the Windows phone but did not discuss the clear experimental evidence
KaKaoTalk [44, 46]	Android	Kakao encrypted database was decrypted to gain access to messages and contact information	Reference [41] discussed the security measures against password guessing attack to protect the encrypted databases. Reference [44] did not analyze the whole file system (part analysis has been done)
Instagram and Path [47]	iPhone 5s	Followers, hashtags, and some account information recovered from Instagram. Path did not disclose any important data of user activities	Discussed and proved the importance of write blocker while acquiring data from a device
Cubby, IDrive [48]	Android v5	Recovered register id and password, timestamps, uploaded files with timestamps, and downloaded files with timestamps and recovered encryption technique	Some locations were identified that were vulnerable to attacks

TABLE 1: Continued.

App	Platform used	Artifacts	Remarks
Facebook, Twitter, Myspace [49]	Android, iPhone, and Blackberry	Recovered some information from iPhone and Android. Nothing recovered from Blackberry	Logical backup copies of three smartphones. No artifacts from internal memory were discussed
Snapchat [33]	Android	Unexpired and expired recoverable	Scenario-based discussion. Scenarios are difficult to implement because of high cost

for forensic analysis. The artifacts suggested by our study can be helpful in forensic investigation of cybercrimes on SNAs.

The rest of the paper is organized in 6 sections. Section 1 presents the preliminary concepts and definitions used in this paper. Section 2 presents the literature related to mobile app forensics on different operating systems (OSs). In Section 3, the methodology of the research is explained. Section 4 covers the artifacts recovered from all five apps using Magnet AXIOM, Autopsy, and XRY. Section 5 discusses the results gathered from three tools and evaluates the tools according to total artifacts recovered, NIST parameters [19, 20], and additional parameters derived during this research to judge the tool's capabilities. Conclusions and future work are presented in Section 6.

2. Preliminary Concepts

This section presents a brief overview of preliminary concepts that are going to be used/referred throughout this article.

2.1. Android Operating System. Currently, Android OS is the most commonly used OS in mobile phones with an 88% share in the worldwide smartphone industry. It is therefore essential to explore Android using various methodologies and methods [21]. For forensic investigators, the folder structure of an Android phone can be an extremely interesting region. So, they should understand where the information/evidence can be found. It is therefore helpful to understand the structure of data storage [22].

A unique Id (UId) is assigned to each app in Android. Each app runs in a separate process so as no application can access the data of other app. A unique app id for a specific app is stored in the app package. Phone application can store app data in many ways [23]. Through app forensic analysis, an investigator can comprehend the usage of the app and find the user data. App analysis is important because nearly all of them use typical function, i.e., messages, calls, contacts, and internet surfing [24]. This data can tell a lot about the user as to when they were in a specific location or to whom they have communicated and their future planning, etc.

2.2. Digital Evidence and Forensic Process. In 2006, Carrier and Spafford [25] defined a digital evidence as the data that approve or disapprove the hypothesis made about the digital events. Forensic investigation is done by collecting, preserving, and analyzing the evidence to present in the court of law. Mobile phones continuously transmit the data through

wifi, Bluetooth, etc. It is too difficult to preserve data without altering it so it is important to record and document every single detail during the whole process.

According to the National Institute of Standards and Technology (NIST), the forensic process [26] includes a 4-step procedure, i.e., preservation, acquisition, analysis, and presentation. Figure 2 describes the NIST forensic process.

2.2.1. NIST Standards on Smartphone Extraction Tools. NIST releases some parameters and methods to calculate the performance of forensic tools based on the outcomes of the assessment plan conducted by NIST. Every assertion creates at least one experiment comprising of a test convention and the normal test outcomes. The test convention indicates point-by-point techniques for setting up the test, executing the test, and estimating the test outcomes [19]. NIST claims that in forensic cases, expanding the quantity of cell phones every year creates problems. Hence, to quantify the capability of forensic tools available, a method is required. NIST offers 42 parameters and methods based on the results of each test plan to assess the performance of forensic devices.

The objective of the computer forensic tool testing (CFTT) venture at the NIST is to build up an approach for testing forensic tools. This is done by establishing unique and common rules governing the requirements of the tools. NIST records the estimation parameters of the forensic tools on two composed reports entitled "smartphone tool specification" [19] and "smartphone tool test assertions and test plan" [20]. The estimation parameters are partitioned into two parts: core and optional. Specifications for smartphone devices are in two parts. Smartphone tool core requirements ([SPT-CR-01] to [SPT-CR-06]) are the requirements that will be met by all the acquisition tools. Smartphone tool optional requirements ([SPT-RO-01] to [SPT-RO-15]) require that all tools shall comply with the requirements that the stated feature or choices that tool offers. Test assertions are developed using these requirements. Test assertions are defined as general proclamations of conditions that can be tested after a test has been carried out.

2.3. Digital Investigation Tools. There are tools that are designed to acquire and analyze the digital image from the mobile devices. Competency of the forensic acquisition and analysis of these tools can differ from one another so it is important for the analyst to have the knowledge of different tool expertise levels. The output comparison and verification between tools can help the examiner to choose the tool he needs to use for the case. We have used the following three tools in our analysis.

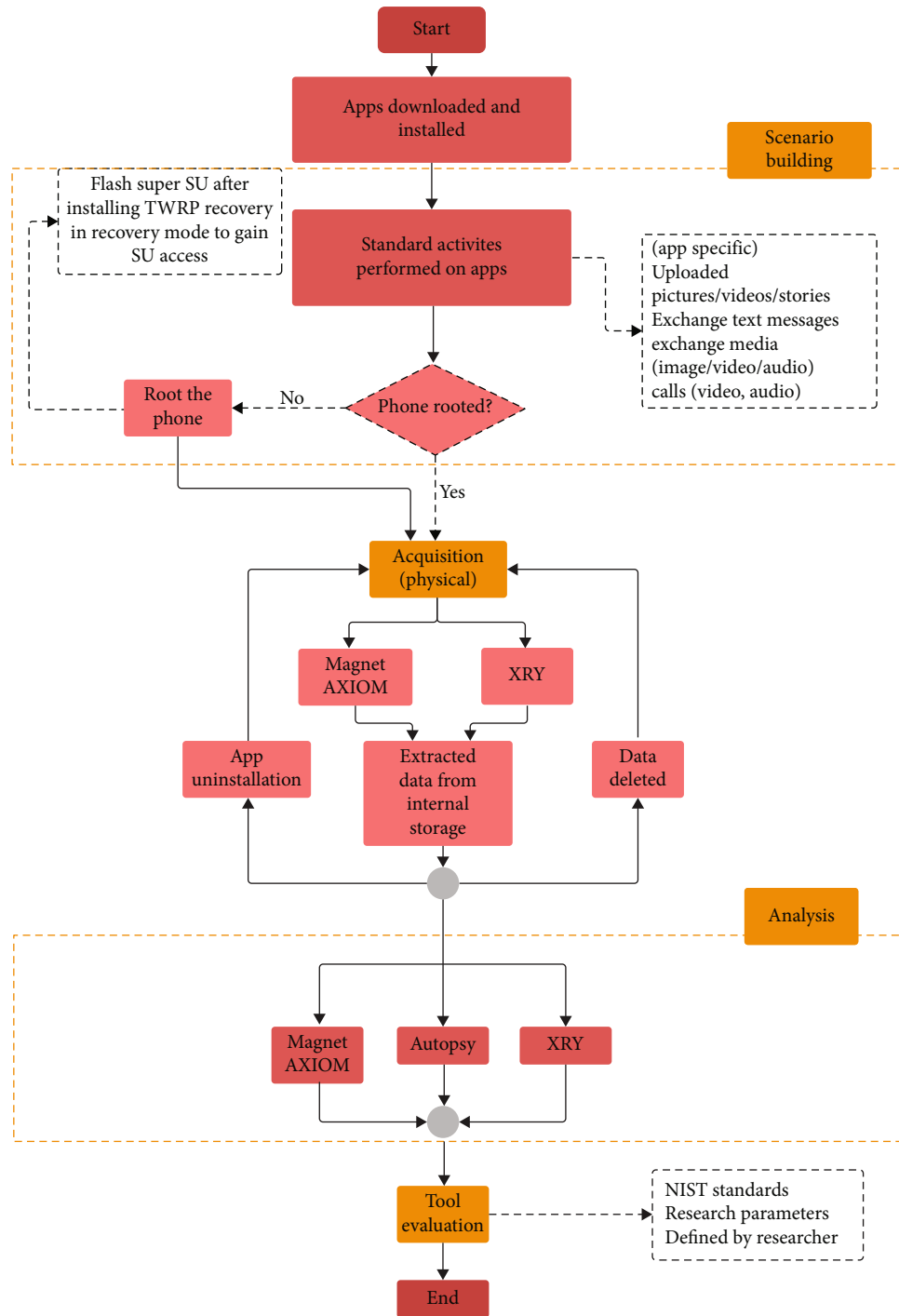


FIGURE 3: Research methodology: a detailed flow chart of actions performed during forensic analysis.

2.3.1. *Magnet AXIOM*. AXIOM is a complete digital investigation tool developed by Magnet Forensics. It is used to recover digital evidence from different sources, i.e., computers, smartphones, third-party images, and cloud. This platform contains two apps to acquire and analyze data. The *AXIOM* process is used to acquire and process the data acquired from the smartphone, and *AXIOM Examine* performs the examination and analysis over the acquire data. For the purpose of this research, we have used the fully functional trial version [16].

2.3.2. *Autopsy*. Autopsy is an open source digital investigation platform that is commonly used by law enforcement and forensic examiners to analyze the digital image in order to get the evidence from it. In this research, we are using Autopsy as a second analyzing tool to make certain of all the evidence recovered from the acquired images [17].

2.3.3. *XRY*. XRY is a digital investigation platform. It is an instinctive and competent software app that runs on the Windows OS. It allows an examiner to extract high-quality

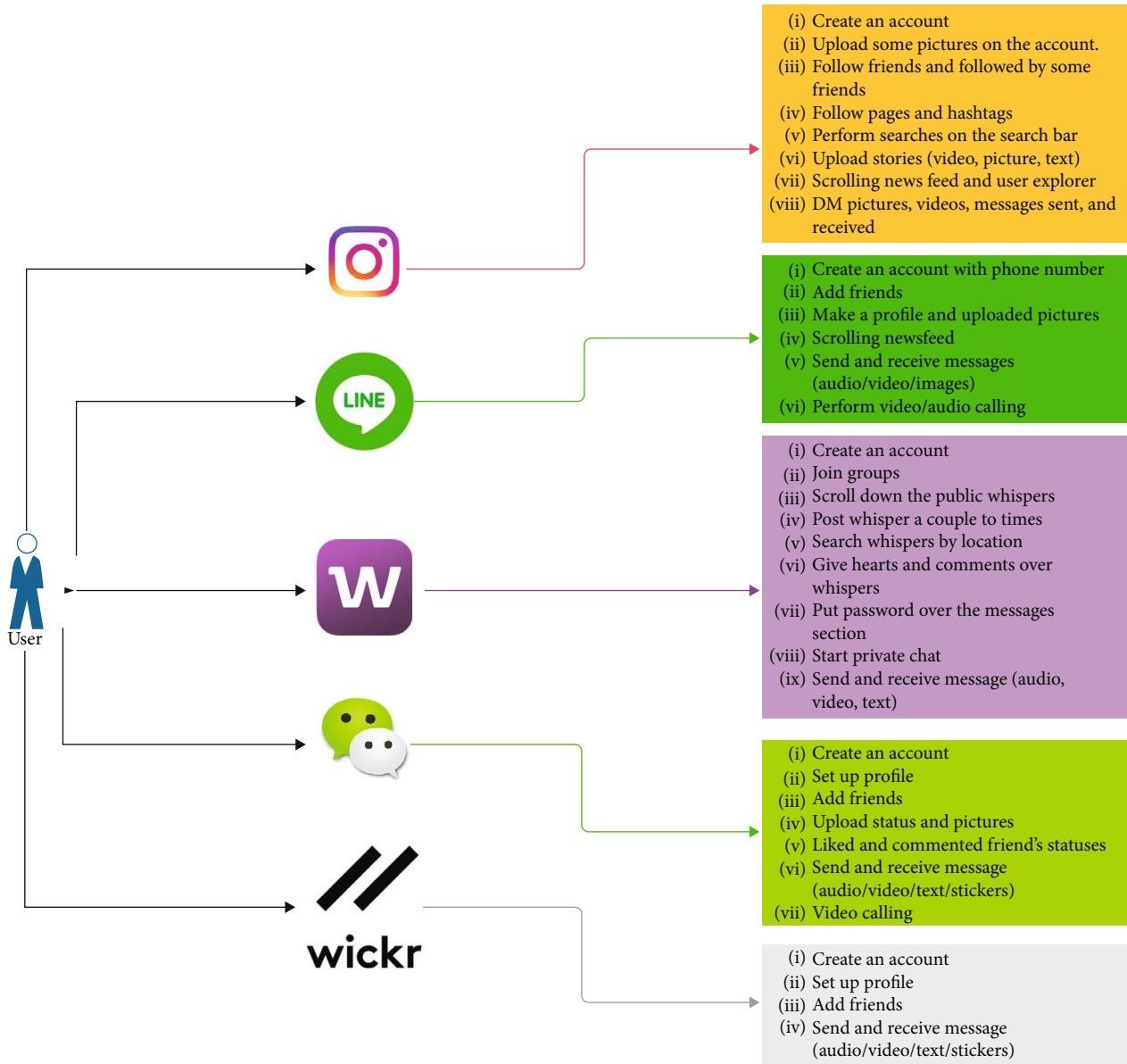


FIGURE 4: Application scenario.

TABLE 2: Experimental tools (SNA stands for social networking app).

Sr.	Experimental tools and software	Description
1	Samsung Galaxy J7	Android 8.0.1
2	Workstation	Windows 10, 64 Bit, Intel i5
3	USB cable	Connect the mobile phone to the workstation
4	SQL DB browser [58]	Version 3.11.0
5	Instagram [11]	SNA v87.0.0.18.99
6	LINE [12]	SNA v9.4.5
7	Whisper [13]	SNA v7.28.6
8	WeChat [14]	SNA v7.0.3
9	Wickr [15]	SNA v5.10.2

data securely from different digital devices and platforms. Acquisition and analysis can be performed through this tool. It allows an examiner to extract logical or physical data according to the case [18].

2.4. Root for Physical Acquisition. Forensic examination requires a detailed recovery of artifacts for thorough analysis, even though rooting is not needed for physical acquirement in some cases where patch is offered by the acquisition tool such as XRY, Cellebrite, and Magnet AXIOM. On the other hand, rooting the device helps in eliminating the limitations that the cell carriers or system OEMs have imposed. A rooted interface offers effective user data extraction and access to internal directories for the device. The partitions and system folders are kept hidden with no access for a nonrooted phone. However, many Android smartphone manufacturers permit to legally root your devices [27]. Moreover, integrity of user data from rooted Android devices during data acquisition is a main concern as forensic analyst extract valuable data from Android phones by rooting [27]. Furthermore, authors in [28] prove that rooting of Android devices has legal validity and the evidence extracted as a result of the rooting process is effective and credible evidence of conviction in criminal proceedings.

3. Related Work: Application Forensics

Some research work has been done in the field of mobile application forensics. Some of the analysis is done on the device general activities, event logs, and device logs [29, 30], whereas others emphasized on the installed applications on the device. Andriotis et al. [31] related the usage of smartphone with numerous crimes like confidential information sharing on public mediums, uploading images over the cloud and child pornography etc. [32]. Information was collected from phone log files, wifi logs, event logs, Bluetooth logs, and databases containing the browsing history. Snapchat was analyzed in [33] by Infosecurity Group and by Aji et al. [34] on two smartphones using Android and iOS. They acquired the data from the smartphone's internal memory through 3 extraction techniques: physical, logical, and file system. Extraction was performed with UFED Cellebrite. Chatting file, images, and videos were detected from XML records found on the iOS smartphone; however, the Android device data was not permanently deleted but hidden with nomedia extension.

In [35], Mehrotra et al. aimed to authenticate the founder's claim that the Android application Wicker enables the user to exchange self-destructive messages and files. They examined both rooted and not rooted Android phone data acquired through *Titanium Backup Android app v6.1.1* and *Helium Backup Android app*. No artifacts or trace of data exchange was found. Mahajan et al. [36] analyzed the artifacts of two apps, *WhatsApp* and *Viber*. Data was extracted through *UFED* from 3 versions of Android OS. Both the apps were examined through *UFED Analyzer*. The chat list, chat messages, and sessions along with timestamps were found in the WhatsApp "msgstoredb" file, and contact information was found in the "wadb" file. For *Viber*, all information that

TABLE 3: Forensic tool description.

Sr.	Forensic tools	Version	Description
1	Magnet AXIOM	4.0	Proprietary
2	Autopsy	4.14.0	Open source
3	XRY	8.2	Proprietary

TABLE 4: Proposed categories for artifacts.

Categories	Artifacts
DB	Databases recovered
	Text messages
Media/text exchanged	Images incoming
	Images outgoing, videos incoming, videos outgoing, audio incoming
	Audio outgoing, GIFs/emojis incoming
	GIFs/emojis outgoing
	Stories, posts uploaded
Timeline	Posts liked/reply
	Group information
	Timing of posts/status
	Friend list
Account/user information	Profile picture, date of birth
	Email address
	ID
	Location
	Name
	Phone number
Calls	Profile images
	Phone app activity
Timestamps	Audio calls, video calls
	Private chats
	Stories/posts/statuses upload
	Friends added

include send and receive messages, contact lists, and timestamps was found through manual search. Mathavan and Meeran [37] performed forensic analysis on *WhatsApp*. The internal memory of an Android phone was analyzed to find out the artifacts such as send/receive messages, images, videos, logs, and contact information. Walnycky et al. [38] selected 20 social messaging apps based on the number of downloads and keyword results from Google Play Store. Network traffic was captured and saved by using *Wireshark* and examined through *NetworkMiner* and *NetWitness Investigator*. This research concluded that four apps, i.e., *Snapchat*, *Tinder*, *Wicker*, and *BBM* are secure as they encrypt network traffic through HTTPS encryption using an SSL certificate.

Anglano et al. [39] analyzed *ChatSecure* on Android phones. *UFED Physical Analyzer* was used to analyze data. *ChatSecure* database was decrypted through *LiME*. Messages and media shared during conversation were been recovered, whereas the deleted data was not recovered. Adebayo et al. [40] analyzed *Kik* app installed on three Android mobile devices. The device backup was created with *Titanium*

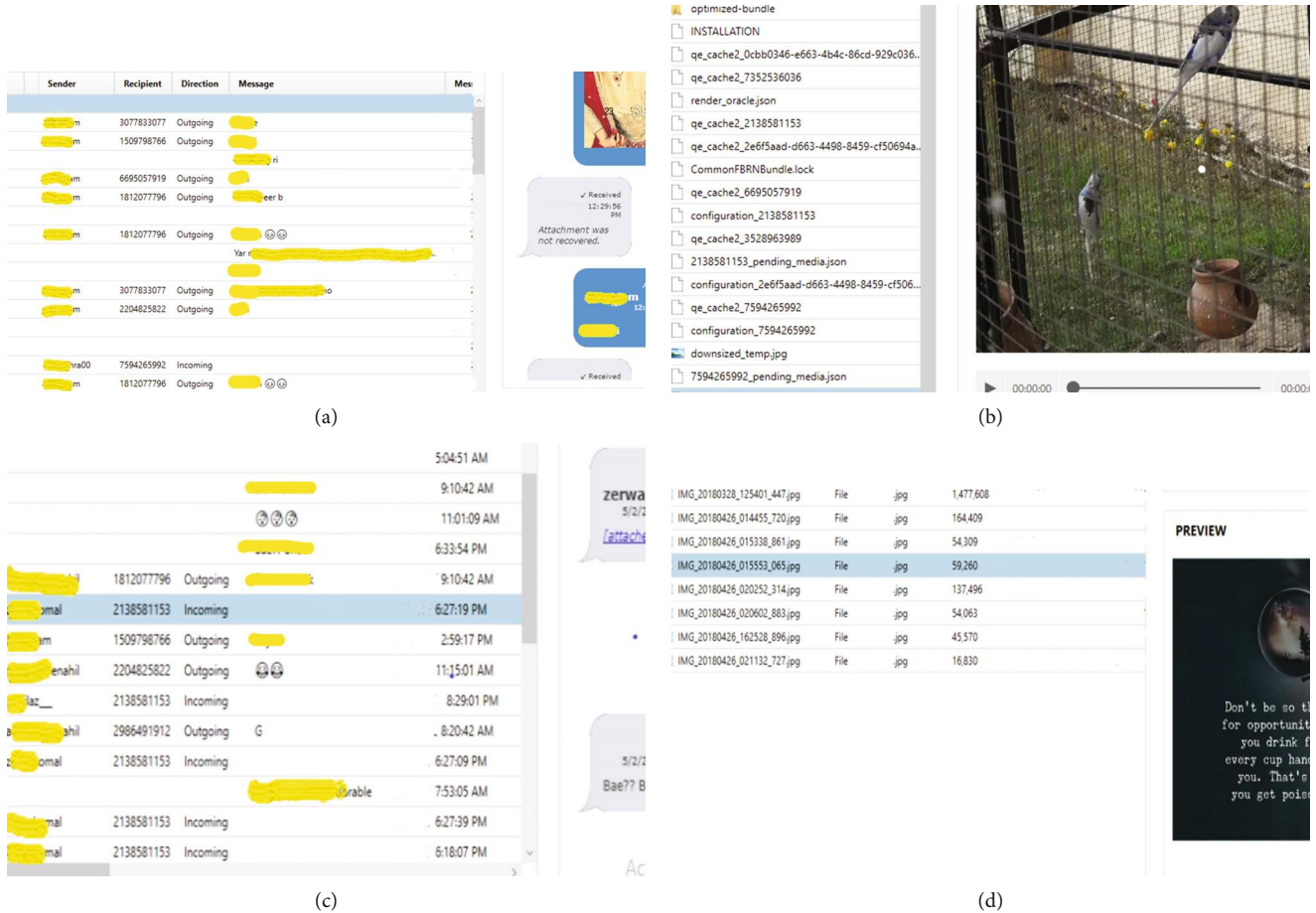


FIGURE 5: Artifacts retrieved from Instagram using Magnet AXIOM.

Backup, and the SQLite DB browser was used to analyze the recovered databases. In another study, Instagram was analyzed by Ryu et al. [41] on iPhone 6s using EditPlus3 Plist, iBackupBot iPhone Backup Extractor, iBackup Viewer, and iPhone Tracker DB. User information, activity history, and application settings were recovered from the iPhone backup file. Umar et al. [42] analyzed WhatsApp for digital evidence. The application was installed on Samsung Galaxy S4 GT-I9500 Android version 5.0.1, and acquisition was done through ADB. For analysis, two tools were used: WhatsApp key/DB Extractor and Belkasoft Evidence. Text messages, images, videos, and documents were recovered. From the results generated by tools, Belkasoft was concluded as best among the two. Telegram app was analyzed in [43, 44] on different versions of Android phones and Windows phones [45]. Android gave messages and cache info after exploring DBs. No package related to the app was found on the Windows phone.

In a few other studies [44, 46], KaKaoTalk was analyzed on Android phones. The Kakao encrypted database was decrypted to gain access to messages and contact information in the researches. Facebook, Skype, Viber, Windows Live Messenger, and WhatsApp were analyzed in [50] on iPhone. The backup contains all the information related to these apps even after uninstallation. In [51], Facebook, WhatsApp, Hike, Viber, and Imo were analyzed on an Android phone. Loca-

tions of artifacts were discussed in this research. In [52], the security mechanisms of WhatsApp, Viber, Tango, Voupi, Forfone, HeyTell, EasyTalk, and WowTalk were discussed when they were installed on Android v2.3 and iOS v2.3. In [53–55], LINE Messenger, BlackBerry Messenger, and IMO Messenger app were analyzed respectively on Android phones and iPhone. The content shared between two parties through private conversation was discussed. Twitter, POF Dating, Snapchat, Fling, and Pinterest were analyzed in [56] installed on Android v5. Message content and account information were discussed. Forensic analysis of Snapchat and Burner was done in [57] on both iOS and Android smartphones. Table 1 summarizes some previous studies on forensic analysis on mobile apps.

4. Methodology

The overall methodology adopted in research comprises four steps. These steps are illustrated in Figure 3.

4.1. Scenario Building. In the first stage, investigation scenarios are set up by performing common user activities on apps. Apps are installed on the phone from Google Play Store. Accounts are created for each app and activities, i.e., pictures/videos uploaded, comments, scrolling over newsfeed, stories uploaded, messages (text/audio/video/images) sent

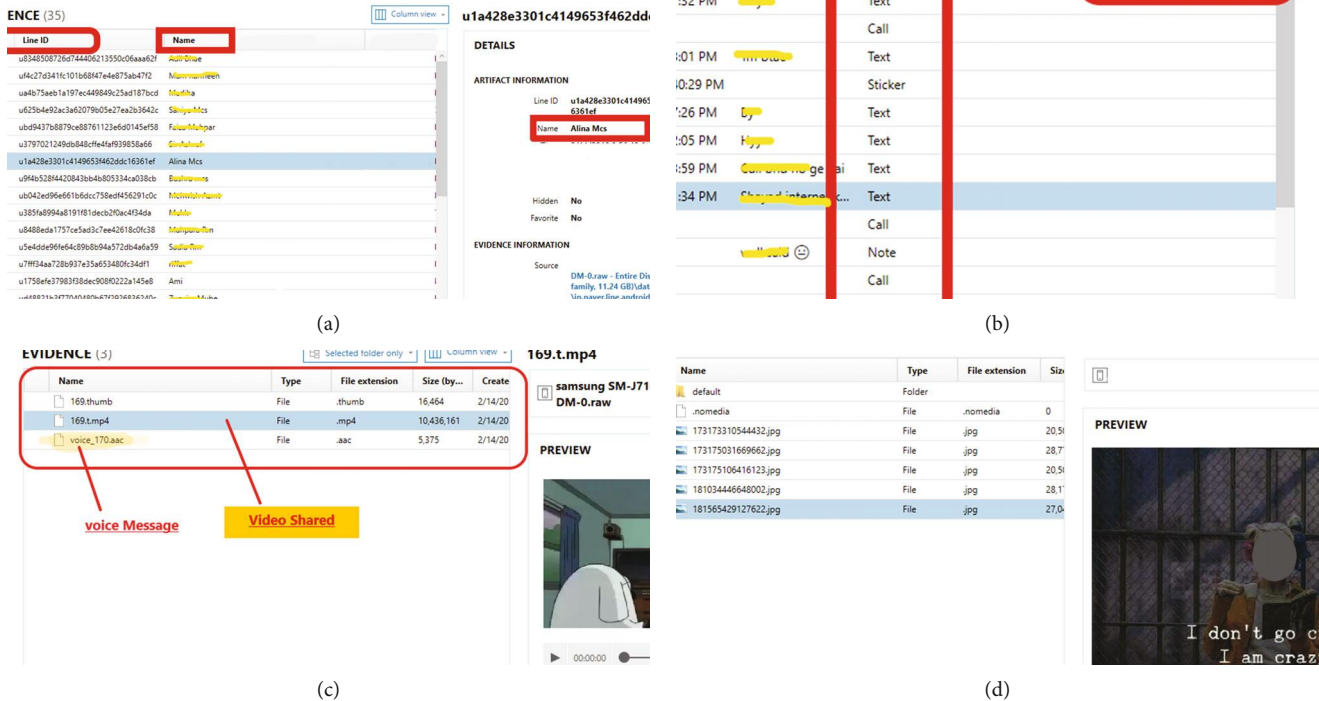


FIGURE 6: Artifacts retrieved from LINE using Magnet AXIOM.

or received, and video calling (LINE), are performed for the application according to their capabilities. The scenario followed in this research is explained in Figure 4.

4.2. *Acquisition.* Data from phone memory is acquired through two different tools *Magnet AXIOM (process)* and *XRY*. Data from phone memory is acquired in three stages.

- (i) Application is installed and working
- (ii) Application is installed and data has been deleted
- (iii) Application and data both have been deleted

In the first stage, all the data remains on the phone as the app is working. In the next stage, some data is deleted by the analyst, and in the last stage, all data is deleted and the app is uninstalled from the phone. Data is acquired from the device in a controlled environment in order to ensure the integrity of the data. In order to get maximum data from the internal memory of the device, data is acquired through the physical acquisition of the device after rooting.

4.3. *Analysis.* In the analysis phase, every app is analyzed by the content of the app folder located in the data/data directory. The analysis generally involves data found in the specific app's file folder and database folder but not limited to them. Another folder found in the data/data/app_folder named "Shared Preference" contains some .xml files having

app-related data. The same information is recovered by all three tools. Cache stores all the activity information and images/videos seen by the user while using the app and is recovered by all three tools.

4.4. *Tool Evaluation.* Tools are evaluated on the basis of their capability to recover digital artifacts from every said app, NIST standards on smartphone extraction tools [19] and some additional parameters from the investigator after conducting research. The result of this research can be used as a recommendation to investigators to handle the cases associated with these apps.

In order to analyze the data generated by these apps (Figure 4), the internal storage of the smartphone is examined after every experiment. The information generated by apps is stored in the inner phone memory that is ordinarily out of reach to users. Therefore, appropriate tools and techniques should be adopted so as to obtain and access this part of the memory. The hardware used in the research is a Samsung smartphone, USB cable, and computer for the retrieval and analysis of data. The description of experimental tools is provided in Table 2. Forensic tools that are used during the experiments are described in Table 3.

Artifacts recovered have been categorized into six fields in this research. The main categories are DB (databases), media/text exchange, timeline, account/user information, calls, and timestamps. The DB category contains the artifacts recovered from the databases present in the app folder. The

Partner Name	Message Text	Status
Hero_Singing	[REDACTED]	Receivec
Twisted Dr	[REDACTED]	Receivec
Twisted Dr	[REDACTED]	Sent
Hero_Singing	Hy	Sent
Twisted Dr	[REDACTED]	Receivec
Twisted Dr	[REDACTED]	Sent
Coolio_Ocean	Hy	Sent
[REDACTED]	[REDACTED]piness	Receivec
Bright_Aluminum	Hy	Sent
Bright_Aluminum	Are you [REDACTED]	Receivec
Bright_Aluminum	Yup	Sent
Bright_Aluminum	[REDACTED]	Receivec
Brave Heart	[REDACTED]	Receivec

(a)

Name	Type	File ext...	Size (b...	Created
2ef4b292-4546-4f5d-a387-1a9d98450661	File		4,142,994	
b229bd4d-6f80-435f-94c1-5dd2e1c43dc	File		3,333,213	

(b)

Name	Type	File ext...	Size (b...	Created
w1517247768051	File		36,929	
w1517820253580	File		62,768	
w1518335770143	File		76,622	
w1518377807121	File		59,507	
w1518378197117	File		103,911	

(c)

ENCE (21)

User Name	Text	Ima
stadium_status	If y	http:
JoeEggs	Just	http:
Cloud_calming	I h	http:
luna_blood	I ju	http:
Stoned_Pallyi	No	http:
seyredhead	Wa	http:
Stoned_Pallyi	At	http:
Jr_Moonlighte	Th	http:
Stoned_Pallyi	Gu	http:
Outlaw_Little	If	http:

Stoned_Pallyi

ARTIFACT INFORMATION

User Name: Stoned_Pallyi

Text: Best time is the time you spend wll your family

Image URL: http://cdn-client.whisper.ca/905644f9bc6200a60ca58187ab0f6cfeaw0bcFe-v5.jpg?v=0

Location: [REDACTED], Punjab, PK

Hearts: 2

Replies: 0

(d)

FIGURE 7: Artifacts retrieved from Whisper using Magnet AXIOM.

FTSSChatRoomMembers Table

FTSSContactLabels Table

FTSSIndexContact Table

FTSSIndexContact_content Table

FTSSIndexContact_docsize Table

FTSSIndexContact_idx Table

FTSSIndexFavorite Table

FTSSIndexFavorite_config Table

FTSSIndexFavorite_content Table

FTSSIndexFavorite_data Table

FTSSIndexFavorite_docsize Table

FTSSIndexFavorite_idx Table

id	c0
18	weixintuandui
19	wxtd
28	friends
29	[REDACTED]
30	[REDACTED] 40
31	[REDACTED] 7
32	[REDACTED] 7
33	friends
34	[REDACTED]
35	[REDACTED]

(a)

FTSSIndexFriend_content Table

FTSSIndexFriend_data Table

FTSSIndexFriend_docsize Table

FTSSIndexFriend_idx Table

FTSSIndexGame Table

FTSSIndexGame_config Table

FTSSIndexGame_content Table

FTSSIndexGame_data Table

FTSSIndexGame_docsize Table

FTSSIndexGame_idx Table

FTSSIndexMessage Table

FTSSIndexMessage_content Table

FTSSIndexMessage_data Table

FTSSIndexMessage_docsize Table

FTSSIndexMessage_idx Table

FTSSIndexMiniGame_content Table

id	c0
4	[REDACTED]
5	[REDACTED]
6	msg_mil_dhy_2
7	[REDACTED]
8	[REDACTED]
9	karre-wxsp00918 ha
10	[REDACTED]
11	[REDACTED]
12	[REDACTED] haggoo yahan se
13	[REDACTED]
14	[REDACTED]
15	[REDACTED]
16	[REDACTED]
17	[REDACTED]
18	[REDACTED]

(b)

VIDENCE (1)

Name	Type	File extension	Size (b...
video_send_preprocess_tmp_1518695915279.mp4	File	.mp4	1,734,114

video_send_pr

FILE DETAILS

File name: [REDACTED]

File extension: .m

Logical size: 1

Created: 2/

Accessed: 2/

Modified: 2/

File attributes: N

VIDENCE INFORMA

Source: [REDACTED]

Textual information of video shared during chat

(c)

Name	Type	File extension	Size (b...	Create
.nomedia	File	.nomedia	0	
msg_42170202151819a6b51fff9104.amr	File	.amr	2,627	

Textual information of audio shared during chat

(d)

FIGURE 8: Artifacts retrieved from WeChat using Magnet AXIOM.

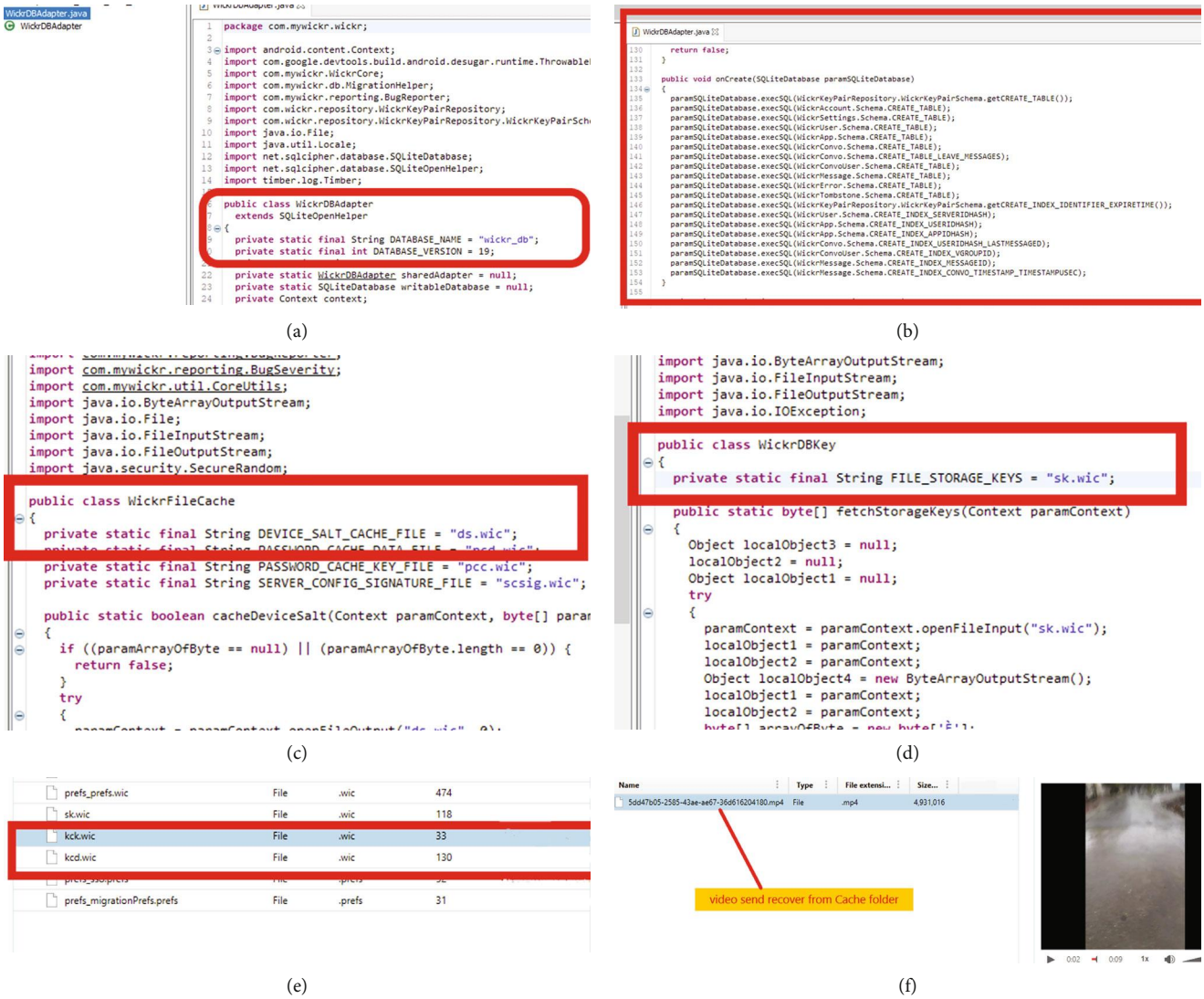


FIGURE 9: Artifacts recovered from Wickr using Magnet AXIOM.

artifacts recovered are related to the exchange of media (images/video/audio/emoji/GIFs) and text between two parties which reside in the media/text exchange category. The timeline category have artifacts related to the information of user’s timeline, i.e., his stories/posts/likes/replies/statuses. Artifacts for the user’s account (profile picture/DoB/email address/ID/name/phone numbers/app activity) reside in account/user information category. The category calls contains the artifacts related to information of audio/video calls done or received by the user. The category timestamps comprises the artifacts related to the timings of different activities performed by the user. The summary of categories has been stated in Table 4.

5. Forensic Analysis

In order to execute the forensic analysis, apps are downloaded from Google Play Store and a set of activities is performed on apps following certain test cases that any user

might perform on these apps. Figure 4 states the activities performed on each app. Physical data of the device is acquired through two propriety tools, i.e., Magnet AXIOM and XRY. Before starting the acquisition, the phone is rooted through the installation of *TWRP Recovery and Flash SuperSu* in the recovery mode. After getting the Super User privileges, full image extraction is performed through Magnet AXIOM and physical acquisition is done by XRY.

The finding of the apps from the acquired image is described in this section. All the activities that are performed by the apps and the relevant data that is stored in the internal memory of the phone is examined. The examination is done by viewing the acquired image through the tools (Magnet AXIOM and XRY) and is analyzed against the defined cases of all the SNAs one by one in detail.

5.1. Forensic Analysis of Apps through Magnet AXIOM. This section discusses the artifacts recovered from the applications using Magnet AXIOM.

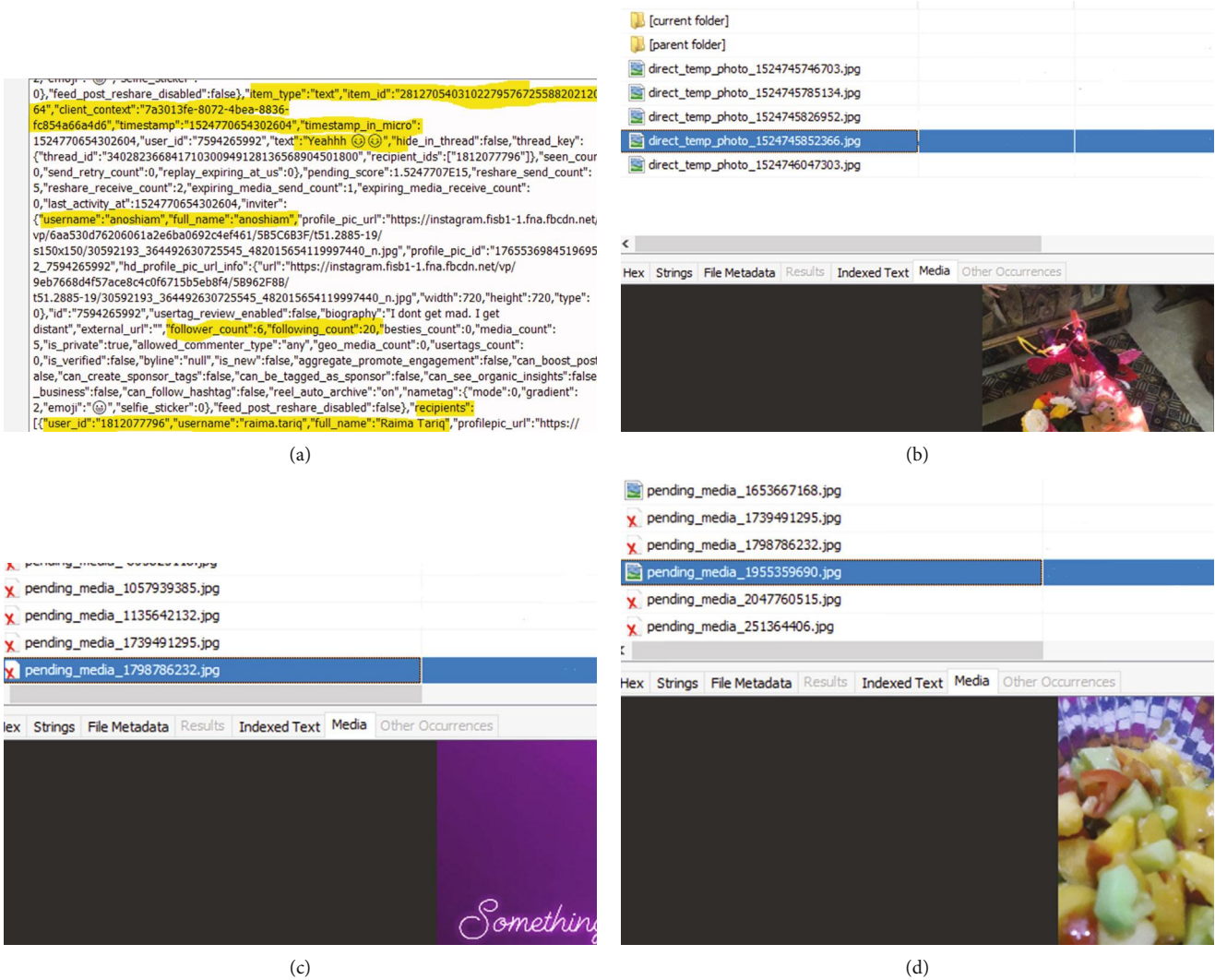


FIGURE 10: Artifacts retrieved from Instagram using Autopsy.

5.1.1. *Instagram*. The artifacts recovered from Instagram through Magnet AXIOM are described in Figure 5 under three conditions: before data deletion, after deleting some data, and after app uninstallation. Firstly, the app is analyzed while no data has been deleted from the device. Figure 5(a) shows that the messages sent and received are recovered with message time, type, sender, and receiver information. Figure 5(b) shows that the stories uploaded by the user on its Instagram account are recovered. An image is taken again after deleting some data, i.e., images, text messages, and images/video uploaded on the Instagram account. Figure 5(c) shows that 80 percent of text messages are successfully recovered and only textual information of images and video calls is recovered such as the name of other party and timestamps. After app uninstallation, no data is recovered related to the Instagram app analyzed by Magnet AXIOM.

5.1.2. *LINE*. As we examined naver.line.db, there are 32 tables from which only 6 tables are of interest from the forensic point of view. The contact table has the relation with call_history.db as the user that was making the call can only be rec-

ognized by the contact table. The id of the caller would be matched from the m_id in the contact table. So, the caller can be verified. LINE gives an end-to-end encryption for data. The public key for every contact is stored in e2ee.db in an encrypted format. After deletion, some messages and media shared through private messages were not recovered. Only a few contacts were recovered. No data related to the LINE application was recovered after app uninstallation. The detailed analysis of LINE is shown in Figure 6. Figure 6(a) shows that LINE contacts, m_ids (unique IDs for every contact), messages, and calls shared between both parties with timestamps are recovered from the database (naver.line.db) store in the app package. Messages recovered with the sender/receiver and message type with timestamps are shown in Figure 6(b). Figures 6(c) and 6(d) show the recovery of media files (videos) and audio image, respectively, transferred during the chat session with timestamps.

5.1.3. *Whisper*. The forensic analysis of Whisper resulted in some data being retrieved. The retrieved data contains

from_mid	content	created_time	delive
j085311ec9e3e3d74ae4c9f5437cbcb5	You registered a phone number.	517919131244	0
je8afca01e0df6d11ac0e2bae09f54473	Welcome to Auto Studio Politeknik official...	517919235296	0
je8afca01e0df6d11ac0e2bae09f54473	NULL	517919235297	0
NULL	Call History : 0 milliseconds, Result: 16	517919575327	1517919570
NULL	Hy	517919619016	0
NULL	Hy	518290631976	1518290625
NULL	Hy	518290730546	1518290724
if9c0671b28787b04703dc20587152059	Hey! Thanks for adding me as a friend. Ch...	518290956854	0
j5fb8644832b726834a8f8e987ab9c3b	NULL	518290970663	0
NULL	NULL	518291093809	1518291087
j5fb8644832b726834a8f8e987ab9c3b	Hey! My ...	518291093892	0
NULL	Okay 😊	518291133106	1518291126
if9c0671b28787b04703dc20587152059	Hey ...	518291133158	0

(a)

(b)

(c)

(d)

(e)

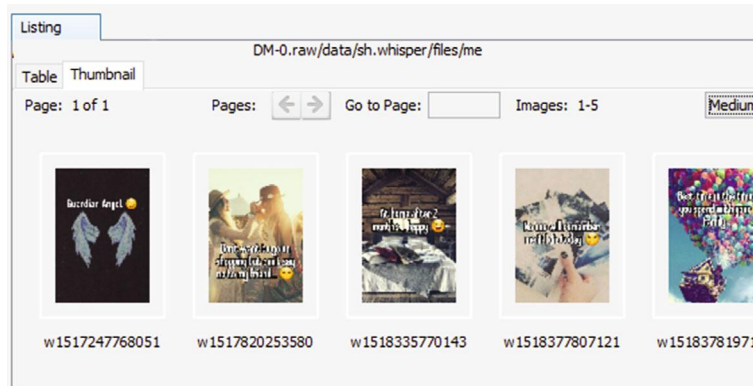
Name	Type	MIME Type	Size
DM-0.raw/media/0/Android/data/jp.naver.line.android	File System	video/mp4	14885

(f)

FIGURE 11: Artifacts retrieved from LINE using Autopsy.

puid	user	ts	url	location	pre	text	ret_text	ima	search	page	image	ate	hearts	
6e36a57d-6a...	Stoned_Pallyi	1517247778480	http://cdn-clie	Somewhere	de.	Guardian Ang...	0	0	NULL	NULL	0	NULL	NULL	2
6e36a57d-6a...	Stoned_Pallyi	1517820264094	http://cdn-clie	Somewhere	de.	Don't want to ...	0	0	NULL	NULL	0	NULL	NULL	1
6e36a57d-6a...	Stoned_Pallyi	1518335779428	http://cdn-clie	Narowal, Punj...	de.	At home after...	0	0	NULL	NULL	0	NULL	NULL	4
6e36a57d-6a...	Stoned_Pallyi	1518377814526	http://cdn-clie	Somewhere	de.	No one will re...	0	0	NULL	NULL	0	NULL	NULL	2
6e36a57d-6a...	Stoned_Pallyi	1518378205078	http://cdn-clie	Narowal, Punj...	de.	Best time is th...	0	0	NULL	NULL	0	NULL	NULL	2
05650f44a5f3...	Jr_Moonlighte	1518606069000	http://cdn-clie	Lake Odessa, ...	de.	That awkwar...	0	0	NULL	NULL	0	NULL	NULL	1564
05652f9e525...	Platinum_Su...	1518629339000	http://cdn-clie	Raleigh, North...	de.	Which fictiona...	0	0	NULL	NULL	0	NULL	NULL	349
05650e8de0...	Whale_Gome...	1518606065000	http://cdn-clie	Escondido, Ca...	de.	I just spent 3...	0	0	NULL	NULL	0	NULL	NULL	1277
05651be5f9e...	stadium_status	1518606070000	http://cdn-clie	Al Muraytibah...	de.	If you could pi...	0	0	NULL	NULL	0	NULL	NULL	443
05650f17219b...	Not.ee	1518606068000	http://cdn-clie	Fond du Lac, ...	de.	Men of quality...	0	0	NULL	NULL	0	NULL	NULL	1081
05651bf53b8e...	will_not_send...	1518606030000	http://cdn-clie	Commerce Cit...	de.	What's one lif...	0	0	NULL	NULL	0	NULL	NULL	413
0564bece9fc...	jheez	1518606064000	http://cdn-clie	Leeds, Englan...	de.	So here I am...	0	0	NULL	NULL	0	NULL	NULL	624
05650f267fc1...	JoeEggs	1518606068000	http://cdn-clie	Preston, Engl...	de.	Just get hoard...	0	0	NULL	NULL	0	NULL	NULL	653
05651be9149...	ponev	1518606069000	http://cdn-clie	Somewhere	de.	My poor fbi a...	0	0	NULL	NULL	0	NULL	NULL	630
05651bf869ed...	lordofkings	1518606071000	http://cdn-clie	Media, Penns...	de.	I don't unders...	0	0	NULL	NULL	0	NULL	NULL	492
05651bd9714...	sexyredhead	1518606069000	http://cdn-clie	Arlington, Tex...	de.	Walked out to...	0	0	NULL	NULL	0	NULL	NULL	539

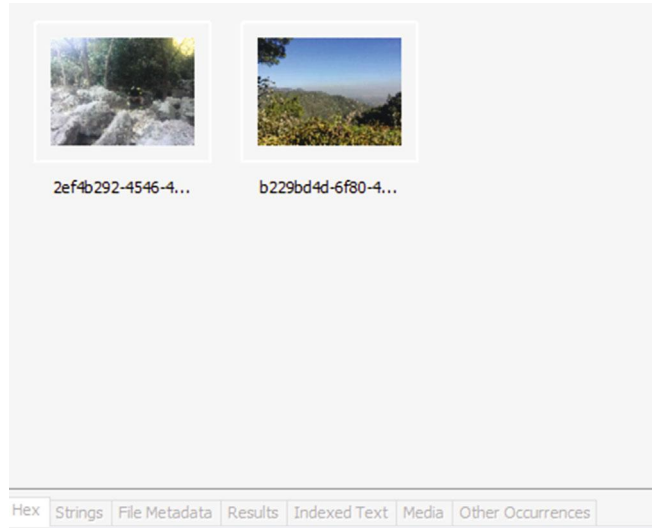
(a)



(b)

sid	gt	wid	im	unread
Stoned_Pallyi	f7cd5caa-193f...	05647311ac9	0000	0
Stoned_Pallyi	7b427077-e3...	05647311ac9	Hy	0
Stoned_Pallyi	2b04ea54-4d...	0564eb18c85	H	0
Stoned_Pallyi	43323422-5df...	0564eb18c85	Top	0
Stoned_Pallyi	f5750d95-cc8...	0564eb18c85	Hi	0
Stoned_Pallyi	4d71fe5a-3ba...	0564eb18c85	?	0
Stoned_Pallyi	6366803a-a2c...	0564eb18c85	C	0
Stoned_Pallyi	33913be3-8f9...	0564eb18c85	M	0
Stoned_Pallyi	a9bbfda2-9b8...	0564eb18c85	M	0
Stoned_Pallyi	e15b966a-59...	0564eb0cb5c	C	0
Stoned_Pallyi	a1a54f8c-d9a...	0564e7b5334	E	0
Stoned_Pallyi	d9c5bad0-b05...	0564e6e1122	V	0
Stoned_Pallyi	e3dc1f22-579...	0564eb18c85	M	0
Stoned_Pallyi	637be4be-99c...	0564eb18c85	M	0
Stoned_Pallyi	92d27e37-8f2...	0564eb18c85	M	0
Stoned_Pallyi	a8c2db3e-f86...	0564f4e2450	M	0
Stoned_Pallyi	eebaf71a-9f6...	0564f4f98c62	M	0

(c)



(d)

FIGURE 12: Artifacts retrieved from Whisper using Autopsy.

information of user accounts, content created or liked by the user, groups he/she follows, private messages shared with friends, location information and other activities. After deletion, all the text messages were recovered and a textual preview of media shared through private messages was recovered. After app uninstallation, no data related to the Whisper app was recovered. The detailed analysis of Whisper

is shown in Figure 7. Figure 7(a) shows the retrieved text messages shared between the user with its friends with timestamps and location information. Textual information of images received by the user is recovered as shown in Figure 7(b). Figure 7(c) shows the retrieved information about the posts uploaded by or replied by the user with timestamps, hearts, and location information. Figure 7(d) shows

Table: FTSSIndexMessage_content

id	Filter
1	[redacted]
2	In [redacted]
3	In [redacted]
4	[redacted]
5	[redacted]
6	[redacted]
7	[redacted] applic
8	[redacted]
9	[redacted] ha
10	[redacted] nai ho r
11	[redacted]
12	[redacted] se
13	[redacted]

(a)

Listing
/img_samsung SM-J710F Full Image - DM-0.raw/media/0/tencent/MicroMsg/71a0c06fe86c

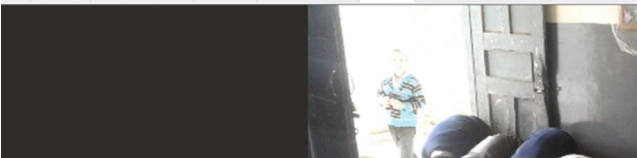
Name	Thumbnail
[current folder]	
[parent folder]	
.nomedia	
msg_01000602111819a6b51aaa6106.amr	[redacted]

audio message recover

(b)

[current folder]
[parent folder]
.nomedia
78d9df08f61fd6b4c7614546b50b7cc6.temp.jpg
th_78d9df08f61fd6b4c7614546b50b7cc6
th_78d9df08f61fd6b4c7614546b50b7cc6hd
th_78d9df08f61fd6b4c7614546b50b7cc6hd_tmp.jpg

Hex Strings File Metadata Results Indexed Text Media Other Occurrences



(c)

com.tencent.mm_preferences.xml
com.tencent.mm_preferences_font_size_report_time.xml
com.tencent.mm_preferences_google_aid.xml
com.tencent.mm_tmp_preferences.xml
com.tencent.mm_webview_x5_preferences.xml
crash_status_file.xml
HalleyServicePreferences_100.xml

Hex Strings File Metadata Results Indexed Text Media Other Occurrences

Page: 1 of 1 Page Go to Page: Script: Lat

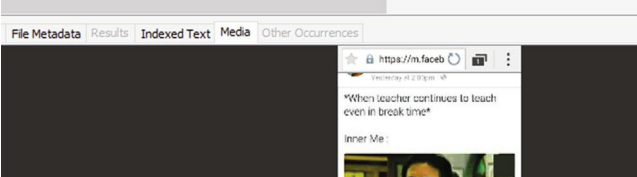
```
<long name="check_trim_time" value="1517924573" />  
<boolean name="hasInitVoiceControlData" value="false" />  
<long name="sql_trace_child_transaction_interval_time" value="5000" />  
<string name="_MTA_DEVICE_INFO_CHECK_ENTIITY_">{"&quot;ts&quot;:1517924573"}</string>  
<string name="last_login_bind_mobile">+9233[redacted]/string>  
<string name="wxid_ng5k5ayccq6c22_sns_entrance_disappear">on</string>  
<long name="handler_log_file_max_size" value="35040" />
```

(d)

d64a868c215805bfa4f2981a925
d64a868c215805bfa4f2981a925hd
d64a868c215805bfa4f2981a925hd_tmp.jpg

picture receive during chat recover after deletion

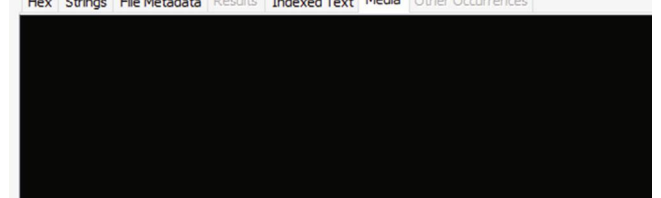
File Metadata Results Indexed Text Media Other Occurrences



(e)

video_send_preprocess_tmp_1518695915279.mp4.h264
video_send_preprocess_tmp_1518695915279.mp4
video_send_preprocess_tmp_1518695915279.mp4.aac

Hex Strings File Metadata Results Indexed Text Media Other Occurrences



(f)

FIGURE 13: Continued.

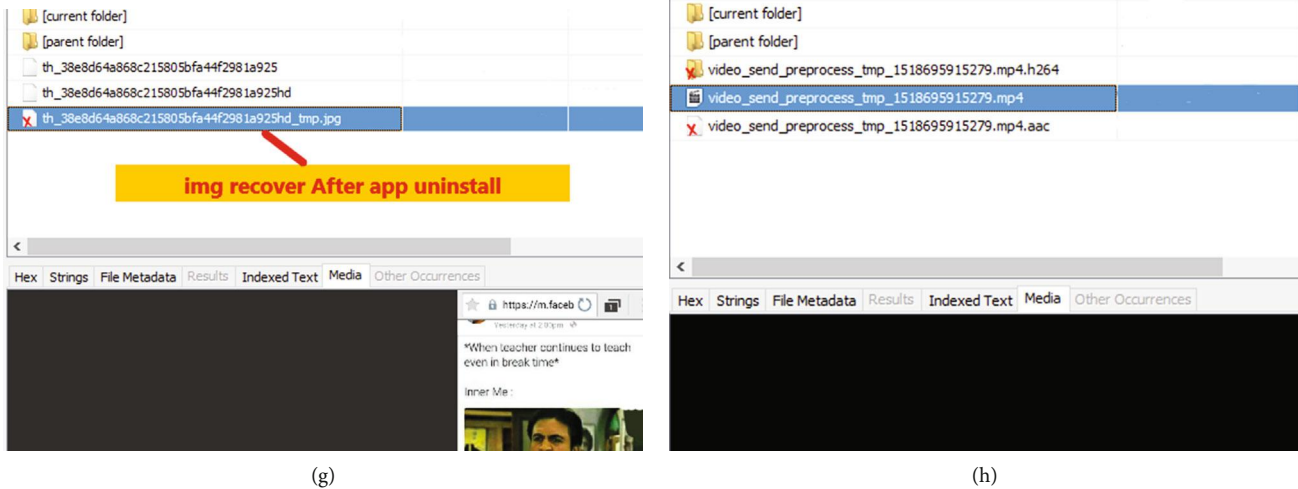


FIGURE 13: Artifacts retrieved from WeChat using Autopsy.

the posts shared while the user was online with timestamps, Whisper content, and location information.

5.1.4. WeChat. WeChat data files are stored within the parent company Tencent's [59] directory MicroMsg folder. WeChat's database EnMicroMsg.db is encrypted using SqlCipher [60]. Some information was retrieved from the index file named FTS5IndexMicroMsg.db, as shown in Figure 8(a). After data deletion, some messages/media files were recovered in a textual format as shown in Figure 8(b). After app uninstallation, only textual information of media files, i.e., video files as shown in Figure 8(c) and audio files (as shown in Figure 8(d)), was recovered from the smartphone. The detailed analysis of WeChat is shown in Figure 8.

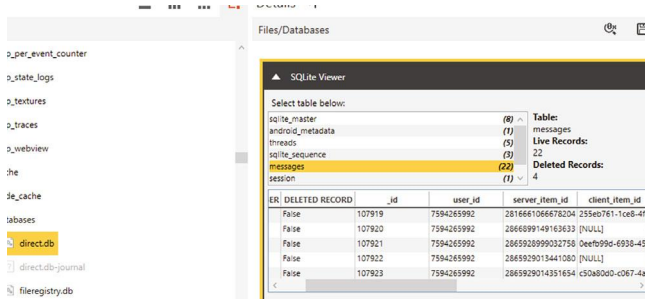
5.1.5. Wickr. Wickr is known as an antiforensic app. It is highly encrypted and claims that no data can be recovered from a device or from network analysis for forensic investigation. All conversations are stored in a highly encrypted database. Wickr does not store any other data within the internal memory of the phone. The detailed analysis of Wickr is shown in Figure 9. By exploring the base.apk file (extracted base.apk file through the .zip archive extractor. In this file, some artifacts have been recovered from classes.dex and classes2.dex files. .dex files were decompiled through Java Decompiler.), it was found that the database wickr_db.db is encrypted with SQLCipher [60] as shown in Figure 9(a). Figure 9(b) shows the WickrDBAdapter File Recover wickr.db Schema that contains account information, contact info, messages sent/received, timestamps, and keys. Figure 9(c) shows that the file folder contains some .wic files that are encrypted. By exploring base.apk, it was found that the ds.wic file is used to store cache data and passwords. Figure 9(d) shows the WickrDBKey.class where it was found that sk.wic contains the key for database. Kck.wic and kcd.wic are also encrypted files that must have contained videos/audio that were sent by the user because these files were deleted after video and audio information sent by the user was deleted (shown in Figure 9(e)). Figure 9(f) shows

that the video sent by the user is recovered from the cache folder.

5.2. Forensic Analysis of the App through Autopsy. This section discusses the artifacts recovered from the apps using Autopsy.

5.2.1. Instagram. Autopsy recovered almost the same data recovered by AXIOM. AXIOM gives the text thread detail with the text/media shared. This is not the case in Autopsy. Autopsy gives the text content and information in the database (have to save the database and open in SQLite browser) shared in DM. The user's activity has been recorded in the cache folder and recovered with timestamps. Video's parts are also recovered in the form jpeg image. After the cache clears, stories and cache data were not recovered. 50 percent of text messages were recovered after deletion from direct.db with date and time information. Images shared during the chat were not recovered. Textual information of media that has been sent or video call done by the user is recovered after data deletion. No data was recovered after app uninstallation. The detailed analysis is shown in Figure 10. Figure 10(a) shows that messages sent through DM are recovered. Figure 10(b) shows that images sent through DM are recovered. Figure 10(c) shows the stories recovered as pending media. Figure 10(d) shows that the images uploaded as the story are recovered after deletion.

5.2.2. LINE. naver_line.db contains the information of text messages shared during private chat, chat history, chat members, and contacts. Autopsy did not recover any text messages after deletion. Deleted images from the timeline have been recovered. Voice message detail recovers after deletion. No data was recovered after app uninstall. The detailed analysis is shown in Figure 11. Figure 11(a) shows the recovered chat history with timestamps and contact information. Figure 11(b) shows that Autopsy recovered the post hidden from the timeline with the option "Hide posts." Backup data of a chat including text messages, media type shared, and call (audio, video) information are recovered as shown in

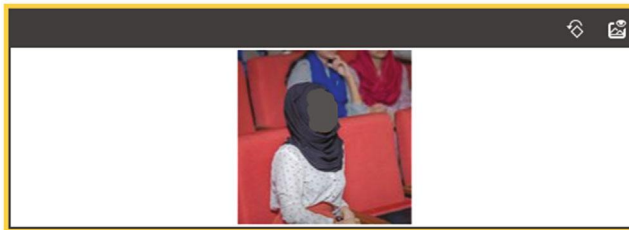


(a)



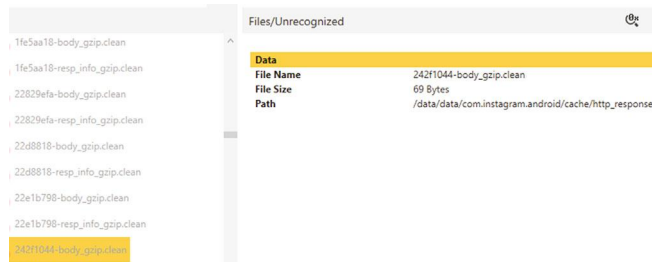
File Name direct_temp_photo_ac4be3d9-fe4d-458d-980b-cb420146
File Format Jpeg
File Size 8.18 KB
Path /data/data/com.instagram.android/cache/original_images

(b)

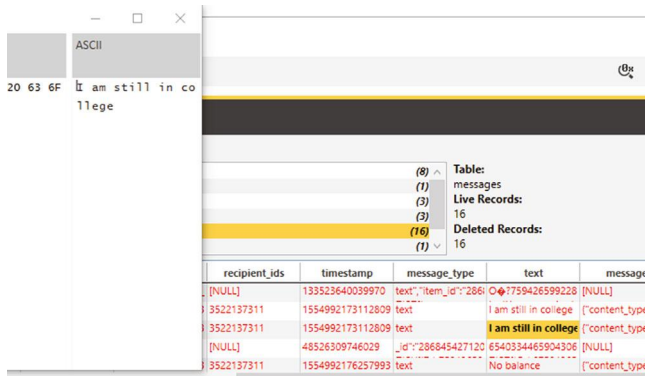


File Name 112179eb.clean
File Format Jpeg
File Size 5.93 KB
Path /data/data/com.instagram.android/cache/images

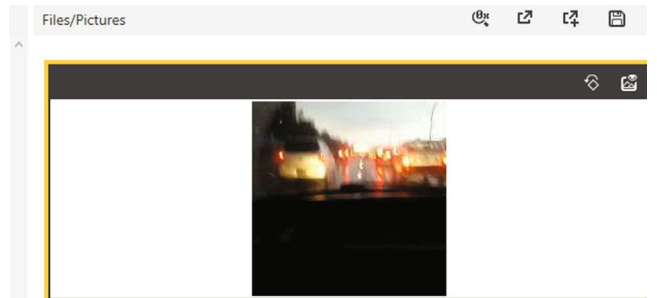
(c)



(d)



(e)



File Name icon_zero_frame.jpg
File Format Jpeg
File Size 2.88 KB
Path /data/media/0/Android/data/com.instagram.android/files

(f)

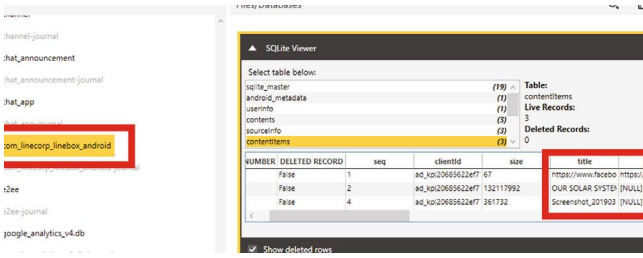
FIGURE 14: Artifacts retrieved from Instagram using XRY.

Figure 11(c). Profile pictures of all the friends have also been recovered as shown in Figure 11(d). Figure 11(e) shows that voice messages were recovered with timestamps. The video shared through chat was recovered in .jpeg image format as shown in Figure 11(f).

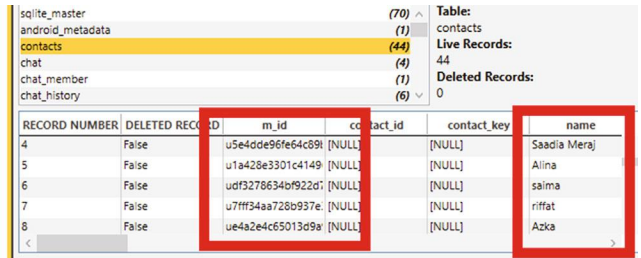
5.2.3. *Whisper*. Artifacts recovered from the c.db database’s c table are the information of private conversation of the user. After deletion, images received while private chatting; Whisper post; event information recovered with the sender name, location, age, gender, and content that has been sent; and some messages with sender and receiver ids and timestamps are recovered after deletion. No artifact was recovered after app uninstallation. The detailed analysis is shown in Figure 12. Figure 12(a) shows the list of every conversation

with the timestamp being stored. The column titled pid has the receiver user id and the column sid contains the user id of the sender. Figure 12(b) shows that Whisper posted on the timeline were recovered. Figure 12(c) shows that Whisper posted by the people while the user was active is recovered with timestamps and locations. Figure 12(d) shows that images received during chat is recovered.

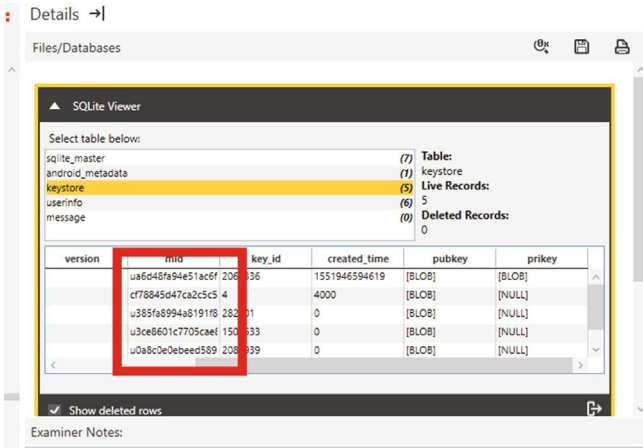
5.2.4. *WeChat*. WeChat artifacts recovered through Autopsy are stated in this section. The detailed analysis is shown in Figure 13. Figure 13(a) shows the messages recovered from the FTS5IndexMessage_content table when it was open in the SQLite browser. Figure 13(b) shows that the audio messages shared during private chat are recovered with timestamps. Figure 13(c) shows that the images received during



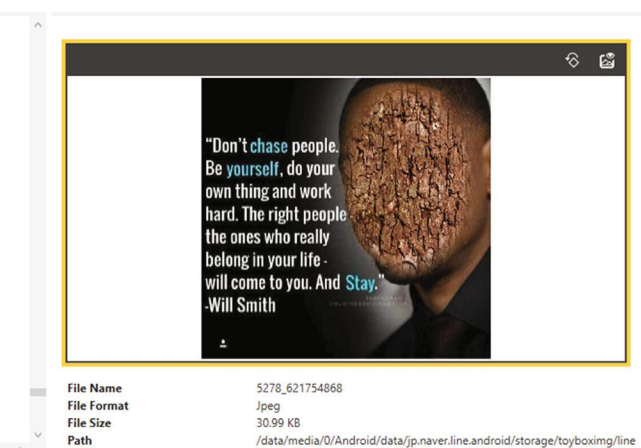
(a)



(b)



(c)



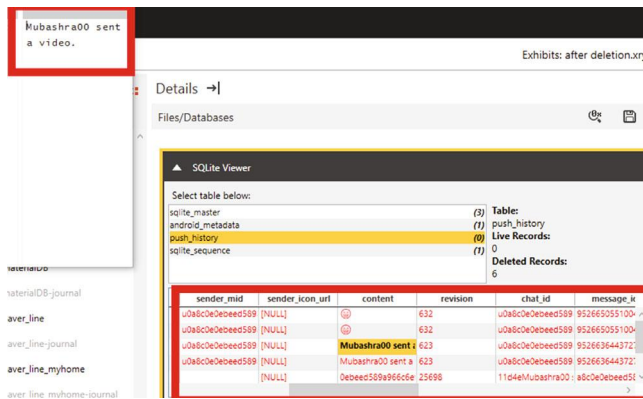
(d)



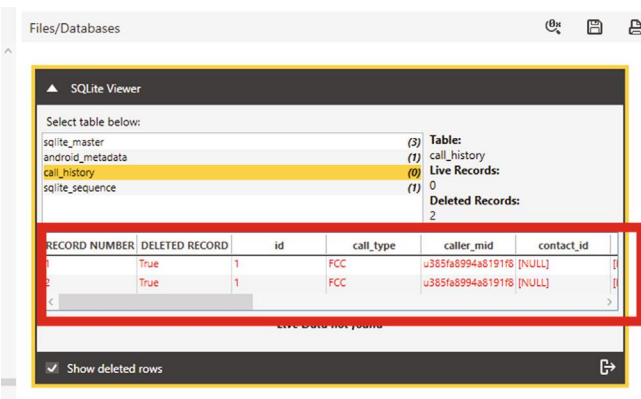
(e)



(f)

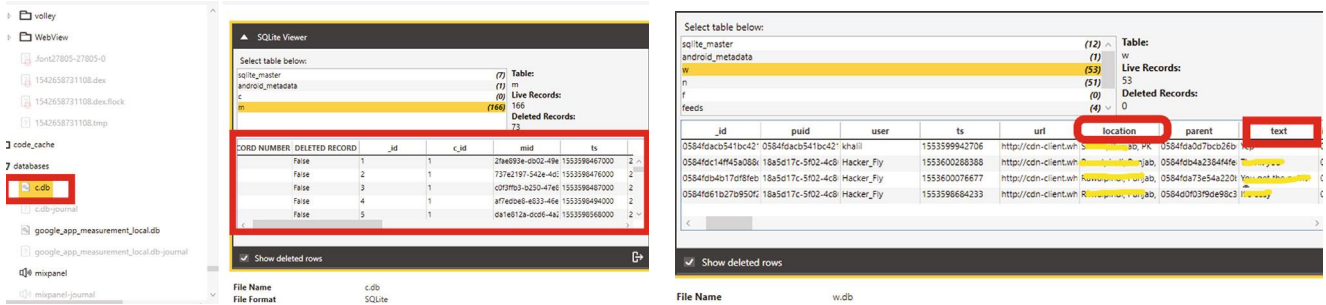


(g)

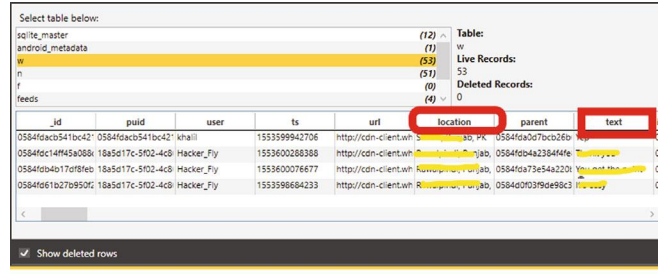


(h)

FIGURE 15: Artifacts retrieved from LINE using XRY.



(a)



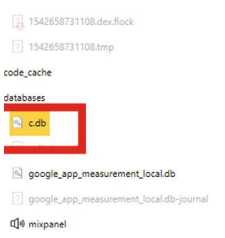
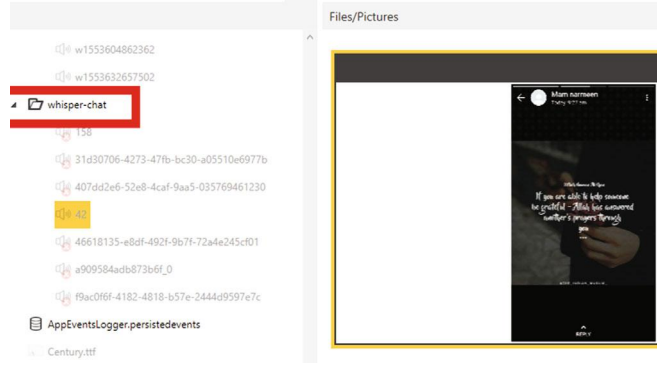
(b)



(c)



(d)

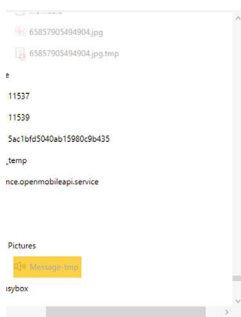


(e)

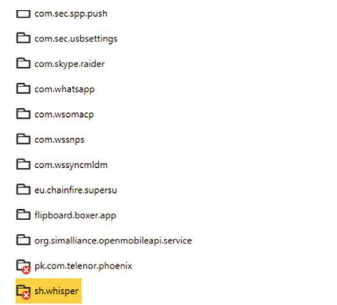
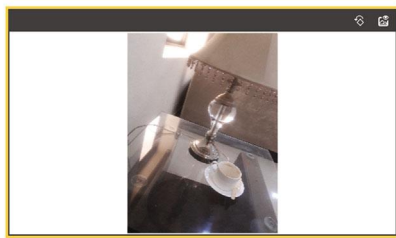
RECORD NUMBER	DELETED RECORD	feed_id	name	display_name	feed_ty
3	False	170bca99-9e89-4ec	House Plants	House Plants	tribe
5	True	8de37d30-7103-40f	Pakistanis	Pakistanis	tribe
6	True	8de37d30-7103-40f	Pakistanis	Pakistanis	tribe

File Name w.db

(f)

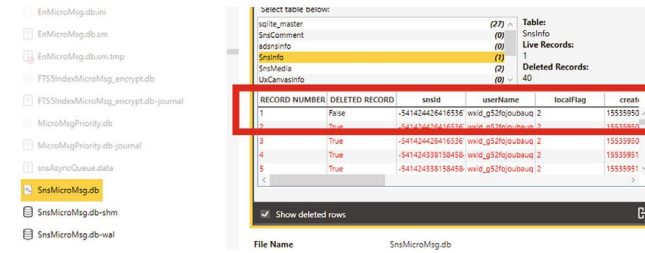


(g)

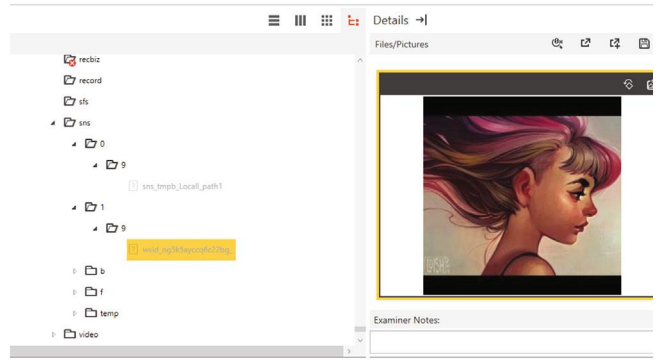


(h)

FIGURE 16: Artifacts retrieved from Whisper using XRY.



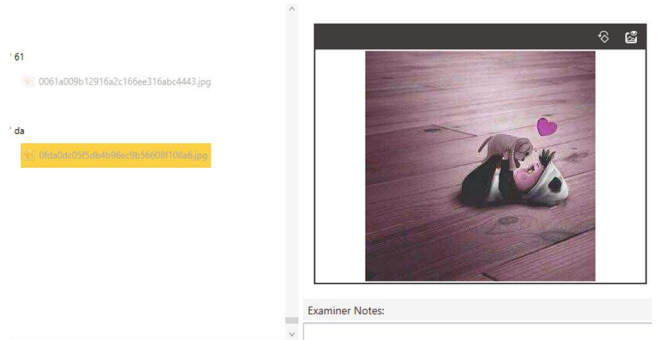
(a)



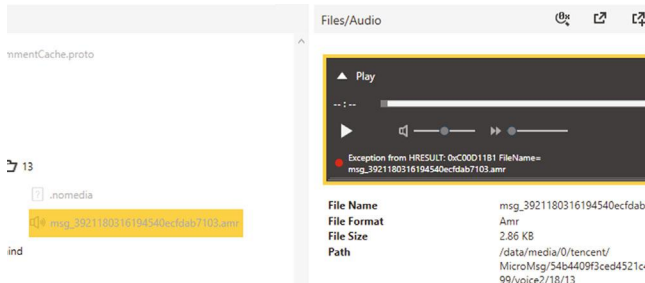
(b)



(c)



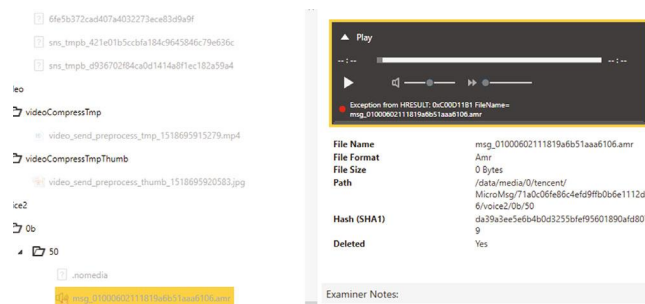
(d)



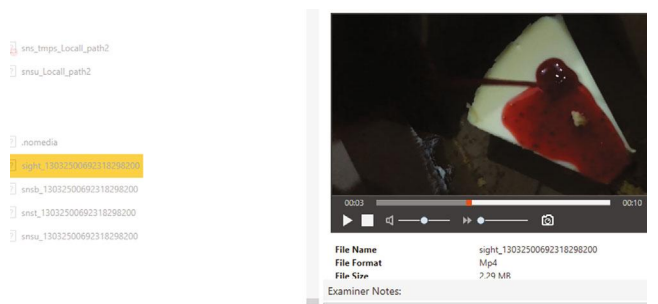
(e)



(f)



(g)



(h)

FIGURE 17: Artifacts retrieved from WeChat using XRY.

chat are recovered. Figure 13(d) shows that the phone number against which the account was recovered in plain text. Textual information of the video shared is recovered after deletion as shown in Figure 13(e). Figure 13(f) shows that images received through private chat are recovered after deletion. Figure 13(g) shows that textual

information of videos is recovered after app uninstallation. Figure 13(h) shows that images shared and uploaded by the user are recovered after app uninstall.

5.2.5. *Wickr*. From analyzing the image from Autopsy, the textual information of the video file is recovered from the

TABLE 7: Location and Artifacts Recovered from Instagram before Data Deletion.

Location	Artifacts recovered	AXIOM	Autopsy	XRY
/data/com.instagram.android/databases/ direct.db	Text messages and sender id with timestamps	(✓)	(✓)	(✓)
/data/com.instagram.android/databases/ direct.db	Text messages received with receiver_id with time stamp	(✓)	(✓)	(✓)
/data/com.instagram.android/databases/ direct.db	http link to the accounts recovered	(●)	(●)	(●)
/data/com.instagram.android/databases/ direct.db-journal	Messages from other accounts previously logged in	(✓)	X	X
/data/com.instagram.android/databases/ direct.db/thread	Every thread info, thread id, sender/receiver ids, account info, messages seen/pending, and type of message	(✓)	(✓)	(●)
/data/com.instagram.android/databases/ direct.db/messages	All the information of message sender, receiver ids, name, type, and content	(✓)	(●)	(✓)
/data/com.instagram.android/databases/direct.db-journal	Send images recovered with date & time and sender/receiver name	(✓)	(●)	(✓)
/data/com.instagram.android/databases/direct.db-journal	Information of received media recovered (type, date time)	(✓)	(●)	(●)
/data/com.instagram.android/databases/direct.db-journal	Link of media shared from newsfeed/explorer through DM	(✓)	(✓)	(✓)
/data/com.instagram.android/databases/video.db	Videos and jpeg thumbnails watched by user on news feed with time/date	(✓)	(✓)	(●)
/data/com.instagram.android/databases/video.db	Videos and jpeg thumbnails watched by user on explorer with time/date	(✓)	(✓)	(●)
media/0/Android/data/com.instagram.android/files/videos/VID_20190319_182811_473_session_0	Video shoot through Instagram cam and uploaded by the user	(✓)	(●)	(✓)
/data/com.instagram.android/files/	Text stories	(✓)	(✓)	X
/data/com.instagram.android/files/	Video stories (playable)	(✓)	(●)	X
media/0/Android/data/com.instagram.android/files/rendered_videos	Video shoot through Instagram cam is being segmented in audio and video files	(✓)	(✓)	(✓)
/data/com.instagram.android/cache/ images	Images viewed by the user on newsfeed with date and time	(✓)	(●)	(✓)
/data/com.instagram.android/cache/ images	Images viewed by the user on explorer with date and time	(✓)	(●)	(●)
media/0/Android/data/com.instagram.android/cache/ExoPlayerCacheDir/videocache	Video information watched by users is recovered	(✓)	(✓)	(✓)

TABLE 7: Continued.

Location	Artifacts recovered	AXIOM	Autopsy	XRY
/data/com.instagram.android/cache/	Stories viewed by a user with date and time	(✓)	(●)	(✓)
/data/com.instagram.android/cache/original_images	Images shared through DM	(●)	(●)	(●)
/data/com.instagram.android/shared_prefs/token_store.xml	Token key recovered	(✓)	(✓)	(✓)
/data/com.instagram.android/shared_prefs/2138581153_insta_video_call_notifications.xml	Live session info of user's followers/following	(✓)	(✓)	(✓)
/data/com.instagram.android/shared_prefs/rti.mqtt.ids.xml	Connection key	(✓)	(✓)	(✓)
/data/com.instagram.android/shared_prefs/rti.mqtt.ids.xml	Device key	(✓)	(✓)	(✓)
/data/com.instagram.android/shared_prefs /com.instagram.android_preferences.xml	Last login_nonce recovered	(✓)	(✓)	(✓)
/data/com.instagram.android/shared_prefs /2138581153_USER_PREFERENCS.xml	Recent user searches recovered	(✓)	(✓)	(✓)
/data/com.instagram.android/shared_prefs/2138581153_usersBootstrapServices.xml	List of the follower and following account recovered with timestamps	(✓)	(✓)	(✓)
/media/0/Pictures/Instagram/	Contain all pictures uploaded by the user	(✓)	(✓)	(✓)
/media/0/Movies/Instagram/	Video uploaded by the user	(✓)	(●)	(●)

cache folder. Wickrdb is encrypted database and all the file folder is encrypted. No information regarding communication has been recovered from Wickr.

5.3. *Forensic Analysis of Applications through XRY.* This section discusses the artifacts recovered from the applications using XRY.

5.3.1. *Instagram.* The image has also been taken from XRY from all the said cases. After the app uninstalls, the app name was recovered with information that the app deleted on what time and date on the data/data folder. The detailed analysis is shown in Figure 14. Figure 14(a) shows that the database direct.db contains the messages and sent images. Figure 14(b) shows that the thread information of the chat through DM is recovered. Figure 14(c) shows the cache data stored in the cache folder of the app package. Figure 14(d) shows that Cache images are recovered in XRY after a cache clears. After deletion, XRY recovered the messages deleted within 24 hrs with text/media information (type, timestamps) as shown in Figure 14(e). Remnants of the video uploaded on the account are recovered after deletion as shown in Figure 14(f).

5.3.2. *LINE.* The LINE package is analyzed before any data was deleted using XRY. After LINE uninstallation, the app existence proof is present in location data/data which contains the application name jp.naver.line.android, date, and time of app deletion. The detailed analysis is shown in Figure 15. Figure 15(a) shows that the files shared through private chat have been recovered. Contact details, chat record, and information shared between two parties including text messages, media, and call info with timestamps were

recovered from naver_line.db as shown in Figure 15(b). Figure 15(c) shows that e2ee.db stores private and public keys encrypted with a unique id and timestamp. Images uploaded by the user on LINE's timeline were also recovered as shown in Figure 15(d). The profile picture of friends was recovered as shown in Figure 15(e). The video uploaded on the timeline by the user was recovered as shown in Figure 15(f). Messages that have been deleted with the content information (text, video, and audio) with sender and receiver ids and timestamps are recovered as shown in Figure 15(g). Figure 15(h) shows that call history details were recovered after deletion from call_history.db images and videos that were uploaded by a user on the timeline are also recovered after deletion.

5.3.3. *Whisper.* Whisper app analyzed by XRY recovered the stated artifacts for all cases. The detailed analysis is shown in Figure 16. Figure 16(a) shows the c.db that contains all the messages that are transmitted between the user and the other users with their ids and timestamps. w.db contains the list of all those people that have posted at the time of the user's connectivity and group user joined as shown in Figure 16(b). Posts uploaded by the user are recovered as shown in Figure 16(c). Images sent by the user were recovered as shown in Figure 16(d). Figure 16(e) shows the cache folder containing the posts viewed by a user with timestamps. Deleted chats are recovered from c.db with the sender and receiver names and timestamps as shown in Figure 16(f). The deleted group's information was recovered. The images sent by the user and then deleted are recovered. The file with the name whisper is recovered after app uninstallation.

5.3.4. *WeChat.* Artifacts recovered from WeChat through XRY is stated below. The first analysis has been done before

TABLE 8: Location and artifacts recovered from LINE and Whisper before data deletion.

Location	Artifacts recover	AXIOM	Autopsy	XRY
/data/jp.naver.line.android/databases/naver_line.db/contact	Contact data recovered with timestamps (m_id, contact_id, name, and status)	(✓)	(✓)	(✓)
/data/jp.naver.line.android/databases/naver_line.db/group_home	group_id and memebtrs_id are recovered	(✓)	(✓)	(✓)
/data/jp.naver.line.android/databases/naver_line.db/group	Group name against their id's is recovered	(✓)	(✓)	(✓)
/data/jp.naver.line.android/databases/naver_line.db/members	0,1 indicated if the requested user accepts to be a member or not	(✓)	(✓)	X
/data/jp.naver.line.android/databases/naver_line.db/chat	Chat_id, message content, and timestamp recovered	(✓)	(✓)	(✓)
/data/jp.naver.line.android/databases/naver_line.db/chat_history	Sender/recover id, message content, date, time, message type recovered (audio, video, text, doc, and location)	(●)	(✓)	(✓)
/data/jp.naver.line.android/databases/naver_line.db/chat_notification	Chat_id	(✓)	(✓)	(✓)
data/media/0/Android/data/jp.naver.line.android/storage/toyboxing/line	User email recover	X	X	(✓)
/data/jp.naver.line.android/databases/naver_line_myhome	Post information uploaded by the ser on LINE timelines	(●)	(✓)	(✓)
/data/jp.naver.line.android/databases/call_history.db	caller_id, time, date, and duration of call (audio, video) recovered	(✓)	(✓)	(✓)
data/jp.naver.line.android/databases/e2ee.db	LINE has end-to-end encryption. E2ee.db stores public keys	(✓)	(✓)	(✓)
data/media/0/Android/data/jp.naver.line.android/storage/mo	Images transferred during chat is recovered with timestamp	(●)	(●)	(✓)
data/media/0/Android/data/jp.naver.line.android/storage/mo	Videos transferred during chat is recovered with timestamp	(✓)	(●)	(●)
data/media/0/Android/data/jp.naver.line.android/storage/mo	Audio messages transferred during chat is recovered with timestamp	(●)	(●)	(●)
data/media/0/Android/data/jp.naver.line.android/storage/write	Images uploaded by the user on LINE timeline recovered with timestamps	(✓)	(✓)	(✓)
/data/media/0/Android/data/jp.naver.line.android/storage/write	Hidden post recovered	X	(✓)	X
/media/0/Android/data/jp.naver.line.android/cache/mm	The video viewed by the user on LINE timeline recovered	(✓)	(●)	(✓)
/data/jp.naver.line.android/shared_prefs/Extended_My_Profile.xml	User's date of birth recovered	(✓)	(✓)	(✓)
/data/jp.naver.line.android/shared_prefs/ServiceInfoManager.xml	Server key for the profile and encryption standard	(✓)	(✓)	(✓)
data/com.whisper.andrid/database/c.db	Private conversion details recovered with timestamps	(✓)	(✓)	(✓)
data/com.whisper.andrid/database/c.db/m	Text messages with timestamps and member_id recovered	(✓)	(✓)	(✓)
data/com.whisper.andrid/database/w.db	Information about the groups joint by the user with time/date and URL	(✓)	(✓)	(✓)
data/com.whisper.andrid/database/w.db/n	Notification received by the user (hearts to whisper/reply to whisper)	(✓)	(✓)	(✓)
data/com.whisper.andrid/database/w.db/w	Whisper posted by people nearby with timestamps	(✓)	(●)	(●)
the /data/sh.whisper/files/me/	Whispers that has been posted by the user (textual preview) recovered	(✓)	X	(●)
/data/sh.whisper/files/whisper-chat/.	Files sent/received by the user to other people recovered	(●)	(●)	(●)
/data/sh.whisper/cache/picasso-cache	All the whisper that has been seen by the user	(✓)	(●)	(✓)
/data/sh.whisper/shared_prefs/com.mixpanel.android.mpmetrics.MixpanelAPI_c39eea2c9ad72a79d1688ca82c50cb94.xml	Whisper app installation data, time, Uid, user location, gender, and date of birth. List of groups that have been joined by the user	(✓)	(✓)	(✓)

TABLE 9: Location and artifacts recovered from WeChat and Wickr before data deletion.

Location	Artifacts recovered	AXIOM	Autopsy	XRY
data/com.tencent.mm/MicroMsg/71a0c06fe86c4efd9b0b6e1112d906/EmMicroMsg.db	Encrypted database. Did not reveal any information	(✓)	(✓)	(✓)
/data/com.tencent.mm/shared_prefs/com.tencent.mm_preferences.xml	u_id, device info, username, and login mobile number are recovered	(✓)	(✓)	(✓)
/data/com.tencent.mm/MicroMsg/71a0c06fe86c4efd9b0b6e1112d906/FTS5IndexMicroMsg.db/FTS5MetaMessage	Messages between U_id of contacts and talkers (receiver of user's messages) with timestamps	(✓)	(✓)	(✓)
/data/com.tencent.mm/MicroMsg/71a0c06fe86c4efd9b0b6e1112d906/FTS5IndexMessagecontant	Text messages recovered	(✓)	(✓)	(✓)
/media/0/tencent/MicroMsg/71a0c06fe86c4efd9b0b6e1112d906/voice2/0b/50/msg_01000602111819a6b51aaa6106.amr	Audio sent by the user is recovered	(✓)	(●)	(✓)
/media/0/tencent/MicroMsg/71a0c06fe86c4efd9b0b6e1112d906/video/videoCompressTmp	Video sent by the user is recovered	(●)	(✓)	(✓)
/data/app/com.mywickr.wickr2-1/base.apk/classes.dex/WickrDBAdapter.class	This file revealed that wickr_db is encrypted by SQLite Cipher (AES-256 full database encryption)	(✓)	(✓)	(✓)
/data/app/com.mywickr.wickr2-1/base.apk/classes.dex/WickrDBAdapter.class	Recovered from WickrDBAdapter.class. DB saves account information, contact info, messages with timestamps, and keys	(✓)	(✓)	(✓)
/data/com.mywickr.wickr2/files	Encrypted .wic files "ds.wic," "sk.wic," Kck.wic, and kcd.wic recovered	(✓)	(✓)	(✓)
/data/app/com.mywickr.wickr2-1/base.apk/classes.dex/WickrDBKey.class	WickrDBKey.class described that "sk.wic" stores the encrypted key of database	(✓)	(✓)	(✓)
/data/com.mywickr.wickr2/files	Kck.wic and kcd.wic must have contained media files (images, video, and audio) because these files were deleted automatically after media of Wickr deleted	(✓)	(✓)	(●)

TABLE 10: Artifacts recovered from Magnet AXIOM after data deletion and app uninstallation.

	Instagram	LINE	Whisper	WeChat	Wickr
	80% of text messages deleted	Call history	Text messages (send/received) recovered	50% of text messages	Encrypted database
	Textual information of shared media	Group member detail	Hearts and replay notification	Image uploaded on the timeline	Video from the cache folder
After data deletion	Account information	Textual preview of video uploaded after deletion	Group data	Video uploaded on the timeline	-
	Uploaded pictures recovered	Textual preview of images	Image sent	Video shared during the chat	-
	-	-	Posts by the user	Images sent recovered during the chat	-
	-	An empty folder names LINE	-	Video uploaded	-
After Apps Uninstall	-	-	-	Image uploaded	-
	-	-	-	Image send during the chat	-

TABLE 11: Artifacts Recovered from Autopsy after Data Deletion and App Uninstall.

	Instagram	LINE	Whisper	WeChat	Wickr
	50 % of Text messages deleted	Hidden/deleted image recovered	Text messages recovered	SSID of chat parties	Encrypted database
	Account information	Call history	Heart and replay notification	Image uploaded on the timeline	Video from the cache folder
After Data Deletion	—	Group member detail	Group data	Video uploaded on the timeline	—
	—	Textual preview of the video uploaded after deletion	Image send	Video shared during the chat	—
	—	Textual preview of images	Posts by user	Images shared during the chat	—
	—	—	—	Image captured by WeChat	—
	A filename with deletion date/time	—	—	Profile picture	—
After app uninstallation	—	—	—	Video uploaded	—
	—	—	—	Image uploaded	—
	—	—	—	Private chat image	—
	—	—	—	Image sent during the chat	—

data deletion. No message information has been recovered as all the databases are encrypted. The detailed analysis is shown in Figure 17. After the deletion of some data, no text messages were recovered. Images and videos uploaded were recovered after deletion. After app uninstallation, all the media files and their information were recovered. Encrypted databases and cache files were not recovered. The profile picture, images, and videos uploaded and shared through private chats are all recovered.

5.3.5. *Wickr*. No data recovery from Wickr could be managed except from the metadata from the base.apk file. After app uninstallation, the location data/data contain the filename com.mywickr.wickr2 with the deleted status being yes.

Detailed information of artifacts that were recovered after three scenarios is discussed in Table 5. The (✓) symbol defines that artifact are recovered from the tool, (●) is the

symbol for textual information/audio-video not playable, and (▲) defines partially recovered. Table 6 describes the artifacts recovered from SNAs before data deletion, after data deletion, and after app uninstallation according to categories proposed in Table 4. The details of artifacts recovered and their location are presented in Tables 7–13. Tables 7–9 state the artifacts recovered from apps and their locations before any data is deleted from apps. Similarly, Tables 10–12 show the artifact information recovered after data deletion and after app uninstallation from every app using all three tools.

6. Results Analysis and Tool Evaluation

This section presents an analysis and discussion on the output of forensic analysis of five SNAs. A comparison of tools on the basis of their capabilities is also presented.

TABLE 12: Artifacts recovered from XRY after data deletion and app uninstallation.

	Instagram	LINE	Whisper	WeChat	Wickr
After data deletion	Text messages deleted within 24 hrs	Group member detail	Text messages recovered	Image uploaded on the timeline	Textual info of the image
	Account information	Call history	Heart and replay notification	Video uploaded on the timeline	—
	Textual preview of the shared image	Video uploaded after deletion	Group data	Video shared during the chat	—
	Video uploaded after deletion	Textual preview of images	Image sent	Images shared during the chat	—
	—	—	Posts by the user	Images received in private chat	—
			Image captured by WeChat	—	
After app uninstallation	Filename with deletion date/time	Filename with deletion date/time	Filename	Profile picture	The app filename recovered
	—	—	—	Video uploaded	—
	—	—	—	Image uploaded	—
	—	—	—	Private chat image	—
	—	—	—	Encrypted database info	—
	—	—	—	App filename	—
	—	—	—	Image sent	Image sent

6.1. Analysis of Apps

6.1.1. Instagram. During investigation of the internal storage of the Android phone for Instagram app data, many artifacts are recovered that can help the investigation. The database folder that contains all the messages, i.e., text message, video, audio, emojis, or the link to online media that have been transferred to or from the user is being recovered unencrypted with the information of date and time. The posts that are uploaded on the account and stories are also recovered with date/time information. The cache folder stores all the online activity done by the user. Every post, picture, story, or video seen by the user get recorded in the cache folder. The shared preference folder contains an .xml file that stored user's account information in plain text format. The number of accounts logged in the app during specific time period are recovered through their login nonce. The information stored in these files contain live sessions attended and last search made etc. After the messages and posts are deleted, data from the database is deleted but some of the messages are being recovered from the .db journal file. The posts have also been recovered after deletion. After the uninstallation of the app, only the pictures that had been uploaded by the user are recovered. If we clear the cache of our phone, photos, videos, and Instagram stories cannot be recovered.

6.1.2. LINE. Different artifacts are recovered from the LINE app. The main focus of our research is on four folders stored inside the local memory of the smartphone. By examining the internal memory of the smartphone, it was determined that the app stores some artifacts on different locations in the app folder. These artifacts are related to its activity within the internal memory. We can understand the DB schema and can recover the critical information of LINE app activity from the database folder. Note that the importance and the

location of the artifacts can be examined during any criminal investigation. It was discovered that LINE manages the directories within its app folder. LINE app stores the cache for transferred, downloaded, and uploaded files in the app cache. From the examination of local memory and databases of the app, we can recover the information (message/media transferred, cache copies) within the DB table in a plain text form with the exception of the password. All the contacts are recovered from the database folder even after deletion. We realize that app data is stored in a different location and different forms can be interesting in a forensic investigation.

6.1.3. Whisper. Whisper post is originally a message sent by user publicly and it includes the name of the sender, message, date/time when it was posted, the link of the image, location, the likes (hearts) it received, and replies to these posts. The name of the user is not a unique identifier in a whisper; that means that the same username can be used by different users and can also be changed anytime by the user. The user phone does not store the images of the post in the app but these images are cached by the phone and stored in the device. Links of posts are also stored on the phone but not all the images are available on these links. Post location information can also be determined in Android with the longitude and latitude for each post. The heart and reply counter keeps the info of likes on a post. Whisper messages are privately sent or received by the user. All the messages with timestamps, media type, and other party information are recovered even after the deletion of some messages or the deletion of the whole thread. Like Whisper posts, in whisper messages, the username is also not a unique identifier so it became difficult to confirm the exact identity of the other party. There is no way to confirm if the message has been received or read by the other person or not. Whisper stores its information against the Mac address of the phone. It does

TABLE 13: Comparison between tools on the basis of the number of artifacts recovered.

Tools	Artifacts	Instagram		LINE		Whisper		WeChat		Wickr			
		Send/receive	Recover	Send/receive	Recover	Send/receive	Recover	Send/receive	Recover	Send/receive	Recover		
Magnet AXIOM	Text messages	Incoming	10	12	10	7	20	20	10	8	5	0	
		Outgoing	10	10	10	8	20	20	10	8	5	0	
	Images	Incoming	5	0	5	0	2	2	4	1	2	0	
		Outgoing	5	5	2	0	2	2	1	1	2	0	
	Videos	Stories	1	1	-	-	-	-	-	-	-	-	
		Uploaded	4	4	4	4	5	5	2	2	-	-	
	Magnet AXIOM	Incoming	-	-	1	0	-	-	1	0	1	0	
		Outgoing	-	-	1	1	-	-	1	1	1	1	
	Autopsy	Text messages	Stories	1	1	-	-	-	-	-	-	-	-
			Uploaded	1	1	1	1	-	-	1	1	-	-
Audio Calls		Incoming	-	-	1	1	-	-	1	1	-	-	
		Outgoing	-	-	1	1	-	-	1	1	-	-	
Video calls		Incoming	1	1	1	1	-	-	1	1	-	-	
		Outgoing	1	1	1	1	-	-	1	1	-	-	
Audio		Incoming	1	0	1	0	-	-	1	0	1	0	
		Outgoing	1	1	1	1	-	-	1	1	1	0	
Autopsy		Text messages	Incoming	10	8	10	8	20	20	10	8	5	0
			Outgoing	10	7	10	6	20	20	10	8	5	0
	Images	Incoming	5	0	5	0	2	2	4	2	2	0	
		Outgoing	5	3	2	0	2	2	1	0	2	0	
	Videos	Stories	1	1	-	-	-	-	-	-	-	-	
		Uploaded	4	4	4	5	5	5	2	2	-	-	
	Audio calls	Incoming	-	-	1	0	-	-	1	1	1	0	
		Outgoing	-	-	1	1	-	-	1	1	1	1	
	Video calls	Stories	1	1	-	-	-	-	-	-	-	-	
		Uploaded	1	1	1	1	-	-	1	1	-	-	
Audio	Incoming	-	-	1	1	-	-	1	1	-	-		
	Outgoing	-	-	1	1	-	-	1	1	-	-		
Audio	Incoming	1	0	1	0	-	-	1	0	1	0		
	Outgoing	1	1	1	1	-	-	1	1	1	0		

TABLE 13: Continued.

Tools	Artifacts	Instagram		LINE		Whisper		WeChat		Wickr		
		Send/receive	Recover	Send/receive	Recover	Send/receive	Recover	Send/receive	Recover	Send/receive	Recover	
XRY	Text messages	Incoming	10	8	10	7	20	18	10	8	5	0
		Outgoing	10	8	10	8	20	18	10	8	5	0
	Images	Incoming	5	0	3	0	2	0	2	2	2	0
		Outgoing	5	0	3	0	2	2	4	4	2	0
		Stories	1	0	-	-	-	-	-	-	-	-
		Uploaded	4	4	3	3	5	5	1	1	-	-
		Incoming	-	-	2	0	-	-	1	0	1	0
		Outgoing	-	-	2	0	-	-	1	1	1	0
	Videos	Stories	1	0	-	-	-	-	-	-	-	-
		Uploaded	1	1	1	1	-	-	1	1	-	-
Audio calls	Incoming	-	-	1	1	-	-	1	1	-	-	
	Outgoing	-	-	1	1	-	-	1	1	-	-	
Video calls	Incoming	1	1	1	1	-	-	1	1	-	-	
	Outgoing	1	1	1	1	-	-	1	1	-	-	
Audio	Incoming	1	0	1	0	-	-	1	0	1	0	
	Outgoing	1	1	1	1	-	-	1	1	1	0	

TABLE 14: Ranking of tools according to digital artifact recovery.

Tools	Instagram		LINE		Applications				WeChat		Wicr		Total recovered artifacts	Index calculated	Ranking
	T	R	T	R	T	R	T	R	T	R	T	R			
Magnet AXIOM	41	37	40	26	49	49	49	49	36	27	18	1	140	76.00%	1
Autopsy	41	28	40	26	49	49	49	49	36	28	18	1	132	71.70%	2
XRY	41	24	40	24	49	43	43	43	36	30	18	0	121	65.70%	3

not have any email id or password and not any phone number needed to register into the platform. However, there is an email registration option that is present in new versions of the app. The user needs to install the app and can start posting and messaging. A username is given by the app. All the whispers can be seen. Turn on the location and the user can see all the whispers that have been posted by the people near its location. If the user uninstalls the app and reinstalls it, it recovers its own account and all the data and activities that have been carried out by the user on that account. But if the user uninstalls the app and restarts its phone and reinstalls the app, its account is gone forever. The user will be registered against another name and gets a new account.

6.1.4. WeChat. From the analysis of the internal memory of the phone, it was revealed that WeChat [14] created the directory Tencent [59] to store its data in the internal memory. The Tencent directory contains all the files including databases, caches, and media information that has been shared or uploaded on the timeline by the user. WeChat [14] cares about privacy more than most social networking apps because of some critical features, i.e., payments. EnMicroMsg.db is a database within this directory which is encrypted. It uses SQLCipher [60] to encrypt its database. All previous researches about WeChat describe that EnMicroMsg.db contains messaging information of the user and describes the method to decrypt this database. A script [61] needs to run with SQLCipher to decrypt the said database. That method is successful for the Android phone's previous versions. For WeChat version 6 or more running on Android version 6.0.x or higher, the database EnMicroMsg.db is not decrypted through the methods described in [61, 62]. Since lower versions of WeChat cannot be installed on Android versions higher than 5, so, the information within this database is not possible to recover. In [62], the previous versions of WeChat were installed on Android 4.4.2 and decrypted successfully using the same methods described in [63, 64]. So, it was determined that the encryption in the latest version of WeChat in Android 6.0.1 or higher is different from the previous version and it is not possible to recover data from it. An index file named FTS5IndexMicroMsg.db contains the information about the contact and plain text messages. The Meta_messages table contains the unique ids of the user; the talker with a timestamp and message_content table contain the content of the messages. The main issue is that we cannot tell who sends which message to whom and when. This database is also encrypted in a later version of WeChat.Media that has been shared through messages and uploaded by the user on the timeline which is recovered from the com.tencent/media/0/MicroMsg folder. It contains the jpeg images, mp4 videos, and audio file transfer during the chat. After data deletion, this folder contains these files stored in this location. The shared preference folder contains critical information, i.e., username and the phone number through which the account was created when the account was created. If we clear the phone cache, it was discovered that the user automatically logs out from the

TABLE 15: Comparison of tools according to NIST standards.

Type	Features	Magnet AXIOM	XRY	Autopsy
Core requirements	SPT-CR-01	(✓)	(✓)	(▲)
	SPT-CR-02	(✓)	(X)	(▲)
	SPT-CR-03	(✓)	(X)	(▲)
	SPT-CR-04	(✓)	(✓)	(✓)
	SPT-RO-05	(✓)	(✓)	(✓)
	SPT-RO-06	(✓)	(✓)	(✓)
Optional requirements	SPT-RO-08	(✓)	(✓)	(✓)
	SPT-RO-09	(✓)	(✓)	(▲)
	SPT-RO-13	(✓)	(✓)	(▲)
	SPT-RO-14	(✓)	(✓)	(X)
	SPT-CA-01	(✓)	(✓)	(X)
	SPT-CA-02	(✓)	(✓)	(▲)
	SPT-CA-03	(✓)	(X)	(▲)
Core assertion	SPT-CA-04	(✓)	(✓)	(✓)
	SPT-CA-05	(✓)	(✓)	(✓)
	SPT-CA-06	(✓)	(✓)	(✓)
	SPT-CA-07	(✓)	(✓)	(✓)
	SPT-CA-24	(✓)	(✓)	(✓)
	SPT-CA-25	(✓)	(✓)	(✓)
	SPT-CA26	(✓)	(✓)	(●)
	SPT-CA27	(✓)	(●)	(●)
	SPT-CA29	(✓)	(✓)	(▲)
	SPT-CA30	(✓)	(✓)	(▲)
	SPT-CA31	(✓)	(X)	(▲)
	SPT-AO25	(✓)	(✓)	(✓)
	SPT-AO26	(✓)	(✓)	(✓)
SPT-AO27	(X)	(●)	(●)	
Optional assertion	SPT-AO31	(✓)	(✓)	(●)
	SPT-AO37	(✓)	(●)	(✓)
	SPT-AO38	(✓)	(●)	(✓)
	SPT-AO39	(●)	(●)	(●)

account and he needs to log in again with the username and password.

6.1.5. Wickr. Wickr secures the internal information by encrypting the local storage. Wickr's delicate information, i.e., id keys, account data, and messages, is stored in an encrypted storage in the phone. This information in the storage container is decrypted only when the user is logged in the account and can be used for any activity. When a user logs off, the container is encrypted again with Klds and expelled from the persistent memory. Klds is put away in an encoded configuration with the goal that it might be recuperated upon the following client login. The key used to encode and decode Klds is taken from the client's passphrase utilizing script [65]. Klds is put away in an encoded configuration with the goal that it might be recuperated upon the following client login. The key used to encode and decode Klds is taken from the client's passphrase utilizing script. Successful login for this

TABLE 16: Comparison of tools according to additional parameters.

Additional parameters	Magnet AXIOM	Autopsy	XRY
Processing time (physical acquisition + processing)	4–8 hrs (fast)		(✓)
	8–12 hrs (medium)	(✓)	
	12–16 hrs (slow)	(✓)	
User friendly	Portable case file	(✓)	X
	GUI	(✓)	(✓)
	Artifact connectivity	(✓)	X
	Categorization	(✓)	(✓)
	DB	(✓)	(✓)
	Media/text exchanged	(✓)	(✓)
Artifacts	Timeline	(✓)	(●)
	Account/user information	(✓)	(●)
	Calls	(✓)	(✓)
Keyword search option	(✓)	(✓)	(✓)
Accuracy	(✓)	(✓)	(●)

situation is equivalent to having the capacity to effectively unscramble Klds and get to an encrypted container material. Those clients who wish to dependably remain signed in to Wickr basically store the secret word-determined key in stage gave secure capacity. Along these lines, delicate material is constantly encoded when the Wickr application is not dynamic [66]. The metadata about how and where Wickr app stores its information is identified while exploring the base.apk file. By exploring the files stored within the .apk file, it was determined that the information related to messages with timestamps and media type is stored in a database named wickr_db. The key through which wickr_db is encrypted is also encrypted and stored within the phone. The database can be decrypted only if the user will log in to the account with the username and password. These phrases with a random number decrypt the key and the key decrypts the database. It was also discovered that a database is encrypted through SQLCipher by the SQL helper class present in the WickrDbAdapter.class file. Two files sk.wic and ds.wic are also encrypted. By analyzing the file in the .apk file of Wickr app, it was discovered that the ds.wic file contains the cache data of Wickr and sk.wic which contain the key of the database which is also encrypted.

6.2. Tools Evaluation. Three tools are used in this research—Magnet AXIOM, Autopsy, and XRY. Magnet AXIOM and XRY have the capability to extract the data from the smartphone and present the artifacts in a human-understandable format. Autopsy just analyzes the already-extracted image. The tools are evaluated on the basis of three factors.

- (i) Number of artifacts recovered by the tool
- (ii) NIST standard tool assessment document [19, 20]
- (iii) Additional parameters

The result of this research can be used as recommendations to investigators to handle the cases associated with

these apps. The overall ranking of tools according to digital artifacts recovery is presented in Table 14.

6.2.1. Number of Artifacts Recovered by a Tool. All three tools are analyzed on the basis of their capability to recover digital artifacts from five SNAs. These numbers give us the validation of the performance of tools. Details of the number of artifacts recovered from every app using these tools are shown in Table 13. Tools are ranked according to the capability of artifacts to be recovered by them. The index number has been calculated according to the formula stated in (1) as follows:

$$P = \frac{\sum Nr}{(Nt)} * 100, \quad (1)$$

where P is the percentage of useful extractions, Nr is the number of recovered artifacts, and Nt is the total number of artifacts

The index number for every application is calculated according to equation (1). The Magnet AXIOM index is calculated by dividing the number of artifacts recovered (R) from all five applications through Magnet AXIOM with a total number of artifacts (T) sent times 100, and the result index is 76 percent. The indexes of Autopsy and XRY are also calculated in the same way. Autopsy is ranked second in as an image analysis tool with an index of 71.7 percent and XRY is ranked third with an index of 65.7 percent.

6.2.2. NIST Standard Tool Assessment Document. NIST published an assessment plan to measure the performance of a tool [19, 20]. It is important to develop the method that can standardize the tool according to its capabilities. NIST releases some factors and methods to calculate the performance of a forensic tool based on the outcomes of the assessment plan conducted by NIST. In Table 15, tools are compared against the core requirements, optional requirements, core assertions, and optional assertions of

smartphone examination tools where the (✓) symbol defines that tool supports the factor, symbol (●) defines partially supporting, symbol (X) defines not supporting, and symbol (▲) defines not applying for a specific tool. According to the NIST parameter of smartphone examination tools, Magnet AXIOM did fulfill most of the requirements.

6.2.3. *Additional Parameters.* Finally, the performance of tools is evaluated on the basis of some parameters that were defined during this research and tool performance capability can be judged by these parameters. These six parameters are

- (i) Processing time
- (ii) User friendliness
- (iii) Compatibility
- (iv) Artifacts recovery
- (v) Keyword search option
- (vi) Accuracy

Table 16 gives a detailed comparison of the tools according to these parameters where the (✓) symbol defines the tool that supports the factor, symbol (●) defines partly supporting, symbol (X) defines not supporting, and symbol (▲) defines not applicable for a specific tool. According to the combined results of defined tool evaluation factors (no. of artifacts a tool recovers, NIST parameter of smartphone analysis tools, and additional parameters) and on the basis of overall performance, Magnet AXIOM is the number one followed by XRY and Autopsy.

7. Conclusions and Future Work

Various tools are available commercially and proprietary through which data acquisition and forensic analysis can be done. In this research, Magnet AXIOM and XRY are used to acquire data from five social networking apps in three different scenarios: before any data deleted from the app, after some data deleted, and after app uninstall. The outcomes of research explain that a large number of artifacts of Instagram, LINE, Whisper, and WeChat are recovered from the smartphone internal memory. Wickr, on the other hand, discloses very little information. Potential artifacts have been categorized to utilize them to create a report. Tools are analyzed with respect to their capabilities, NIST standards for smartphone analysis tools, and few additional parameters defined during this study. The results of this analysis report that among the three tools, Magnet AXIOM is ranked no. 1 with an index no. of 76.0% followed by Autopsy at 71.5% and XRY at rank 3 with an index of 65.5%. According to the NIST parameter of smartphone analysis tools and additional parameters on the basis of overall performance, Magnet AXIOM is the number one followed by XRY and Autopsy. In the future, a new version of Android Smartphones can be analyzed for application forensics as almost every 3 month, a new version or software update is released for Android. This leaves a lot of areas to further research the apps on the latest version and analyze the security flaws.

New apps like *OmeGLE*, *Periscope*, and *Azar* which are becoming popular within teenagers need some attention from forensic investigators. Every tool has some weaknesses, and for the better and accurate results, forensic investigators can use the combination of different tools in his investigation to get more reliable results by using the unique capability of every single tool.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors of this paper are extremely thankful to the Department of Information Security, National University of Sciences and Technology (NUST), Islamabad, Pakistan, for its support in the research.

References

- [1] Global social media research summary September 2019 <https://www.smartinsights.com/social-media-marketing/social-mediastrategy/new-global-social-media-research/>.
- [2] DataReportal – Global Digital Insights, 2021, March 2021 <https://datareportal.com/reports/digital-2020-october-global-statshot>.
- [3] “51 critical cyberbullying statistics in 2020,” <https://www.broadbandsearch.net/blog/cyber-bullying-statistics>.
- [4] “The-Annual-Bullying-Survey-2017-1.pdf,” December 2019 <https://www.ditchthelabel.org/wp-content/uploads/2017/07/The-Annual-Bullying-Survey-2017-1.pdf>.
- [5] N. Jain and V. Shrivastava, “Cyber crime changing everything— an empirical study,” *International Journal of Computer Application*, vol. 1, no. 4, 2014.
- [6] M. Sharma and S. Kaur, “Cyber crimes becoming threat to cyber security,” *IASR Journals*, vol. 2, no. 1, 2019 https://www.academia.edu/39618017/Cyber_Crimes_Becoming_Threat_to_Cyber_Security.
- [7] U. Sharma, S. Ghisingh, and E. Ramdinmawii, “A study on the cyber, crime and cyber criminals a global problem,” *Journal: International Journal of Web Technology*, vol. 3, pp. 172–179, 2014.
- [8] R. A. Grimes, *What Is Personally Identifiable Information (PII)? How to Protect It under GDPR* CSO Online August 2019; <https://www.csoonline.com/article/3215864/how-to-protect-personally-identifiable-information-pii-under-gdpr.html>.
- [9] H. Saini, Y. S. Rao, and T. C. Panda, “Cyber-crimes and their impacts: a review,” *International Journal of Engineering Research and Applications*, vol. 2, no. 2, pp. 202–209, 2012.
- [10] H. Mahalik, R. Tamma, and S. Bommisetty, *Practical Mobile Forensics*, Packt Publishing Ltd, 2016.
- [11] Instagram, App Store, 2010, August 2019. <https://apps.apple.com/us/app/instagram/id389801252>.

- [12] LINE Free Calls, *Messages-Apps on Google Play*GooglecomAugust 2019 <https://play.google.com/store/apps/details>.
- [13] *Whisper Apps on Google Play*GooglecomAugust 2019 <https://play.google.com/store/apps/details>.
- [14] *WeChat Apps on Google Play*GooglecomAugust 2019 <https://play.google.com/store/apps/details>.
- [15] *Wickr Me Private Messenger Apps on Google Play*GooglecomAugust 2019 <https://play.google.com/store/apps/details>.
- [16] Magnet AXIOM Digital Investigation Platform|Magnet Forensics Magnet Forensics August 2019 <https://www.magnetforensics.com/products/magnet-axiom/>.
- [17] *Autopsy*Sleuthkit.orgAugust 2019 <https://www.sleuthkit.org/autopsy/>.
- [18] MSAB, *MSAB The Pioneers of Mobile Forensics*MSABAUGUST 2019 <https://www.msab.com/>.
- [19] R. P. Ayers, *Smart Phone Tool Specification*|NIST, Computer Forensic Tool Testing, 2017, August 2019. <https://www.nist.gov/publications/smart-phone-tool-specification>.
- [20] NIST, *Smart Phone Tool Test Assertions and Test Plan*, NIST, US Department of Commerce, 2010.
- [21] *Android (Operating System)*, Wikipedia, 2019.
- [22] A. Hoog, *Android Forensics-Investigation*, Syngress, 2011.
- [23] "Application sandbox|android open source project," 2019, September 2019 <https://source.android.com/security/app-sandbox>.
- [24] B. Reeves, J. Bowers, S. A. Gorski Iii et al., "droid: assessment and evaluation of android application analysis tools," *ACM Computing Surveys (CSUR)*, vol. 49, no. 3, pp. 1–30.
- [25] B. D. Carrier and E. H. Spafford, "Categories of digital investigation analysis techniques based on the computer history model," *Digital Investigation*, vol. 3, pp. 121–130, 2006.
- [26] R. Ayers, S. Brothers, and W. Jansen, *NIST Special Publication 800-101 Guidelines on Mobile Device*, Obtenido de National Institute of Standards and Technology, 2014, <http://nvlpubs>.
- [27] A. Verma, "What is rooting? Is rooting my android smartphone illegal? Fossbytes," November 2017; <https://fossbytes.com/what-is-rooting-meaning-android-legal>.
- [28] T. Almeahadi and O. Batarfi, *Impact of Android Phone Rooting on User Data Integrity in Mobile Forensics*, IEEE, 2021.
- [29] S. Park, S. Park, and K. Ma, "An automatic user activity analysis method for discovering latent requirements: usability issue detection on mobile applications," *Sensors*, vol. 18, no. 9, p. 2963, 2018.
- [30] F. Rebhi, *Development of a Tool for Analysis and Visualization of Android Logs*, Ecole Nationale des Sciences de l'Informatique (ENSI), Tunisia, 2013, <https://hal.inria.fr/hal-00922034>.
- [31] P. Andriotis, G. Oikonomou, and T. Tryfonas, "Forensic analysis of wireless networking evidence of android smartphones," in *2012 IEEE International Workshop on Information Forensics and Security (WIFS)*, pp. 109–114, IEEE, 2012.
- [32] P. Stephenson, "A comprehensive approach to digital incident investigation," *Information Security Technical Report*, vol. 8, no. 2, pp. 42–54, 2003.
- [33] C. G. Nolte, *Privacy in Social Networks – Economic Options for Regulation*, ResearchGate, 2017, https://www.researchgate.net/publication/324517531_Privacy_in_social_networks_-_economic_options_for_regulation.
- [34] M. P. Aji, I. Riadi, and A. Lutfhi, "The digital forensic analysis of Snapchat application using XML records," *Journal of Theoretical & Applied Information Technology*, vol. 95, no. 19, 2017.
- [35] T. Mehrotra and B. Mehtre, "Forensic analysis of Wickr application on android devices," in *2013 IEEE International Conference on Computational Intelligence and Computing Research*, pp. 1–6, IEEE, 2013.
- [36] A. Mahajan, M. Dahiya, and H. Sanghvi, "Forensic analysis of instant messenger applications on android devices," 2013, <http://arxiv.org/abs/13044915>.
- [37] T. Mathavan and M. A. Nagoor, "Acquisition and analysis of artifacts from instant messenger on android device," *International Journal of Engineering Research & Technology*, pp. 1210–1212, 2014.
- [38] D. Walnycky, I. Baggili, A. Marrington, J. Moore, and F. Breitingner, "Network and device forensic analysis of Android social-messaging applications," *Digital Investigation*, vol. 14, pp. S77–S84, 2015.
- [39] C. Anglano, M. Canonico, and M. Guazzone, "Forensic analysis of the chatsecure instant messaging application on android smartphones," *Digital investigation*, vol. 19, pp. 44–59, 2016.
- [40] O. S. Adebayo, S. A. Sulaiman, O. Osho, J. Alhassan, and S. Abdul-hamid, *Forensic Analysis of Kik Messenger on Android Devices*, ResearchGate, 2017, https://www.researchgate.net/publication/321268908_Forensic_Analysis_of_Kik_Messenger_on_Android_Devices.
- [41] J. H. Ryu, N. Y. Kim, B. W. Kwon, S. K. Suk, J. H. Park, and J. H. Park, "Analysis of a third-party application for mobile forensic investigation," *Journal of Information Processing Systems*, vol. 14, no. 3, 2018.
- [42] R. Umar, I. Riadi, and G. M. Zamroni, "Mobile forensic tools evaluation for digital crime investigation," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 8, no. 3, pp. 949–955, 2018.
- [43] G. B. Satrya, P. T. Daely, and M. A. Nugroho, "Digital forensic analysis of Telegram Messenger on Android devices," in *2016 International Conference on Information & Communication Technology and Systems (ICTS)*, pp. 1–7, IEEE, 2016.
- [44] G. B. Satrya, P. Daely, and S. Y. Shin, "Android forensics analysis: private chat on social messenger," in *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 430–435, IEEE, 2016.
- [45] J. Gregorio, A. Gardel, and B. Alarcos, "Forensic analysis of telegram messenger for Windows phone," *Digital Investigation*, vol. 22, pp. 88–106, 2017.
- [46] J. Choi, J. Park, and H. Kim, "Forensic analysis of the backup database file in KakaoTalk messenger," in *2017 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pp. 156–161, IEEE, 2017.
- [47] R. Al Mushcab and P. Gladyshev, "Forensic analysis of instagram and path on an iPhone 5s mobile device," in *2015 IEEE Symposium on Computers and Communication (ISCC)*, pp. 146–151, IEEE, 2015.
- [48] A. A. Abbasi, S. Saleem, and R. Zulqarnain, "Forensic investigation of smartphone cloud storage applications," *NUST Journal of Engineering Sciences*, vol. 10, no. 1, pp. 30–36, 2017.
- [49] N. Al Mutawa, I. Baggili, and A. Marrington, "Forensic analysis of social networking applications on mobile devices," *Digital Investigation*, vol. 9, pp. S24–S33, 2012.
- [50] Y. C. Tso, S. J. Wang, C. T. Huang, and W. J. Wang, "iPhone social networking for evidence investigations using iTunes forensics," in *Proceedings of the 6th International Conference on Ubiquitous information management and Communication*, ACM, p. 62, 2012.

- [51] V. V. Rao and A. Chakravarthy, "Forensic analysis of android mobile devices," in *2016 International Conference on Recent Advances and Innovations in Engineering (ICRAIE)*, pp. 1–6, IEEE, 2016.
- [52] S. Schrittwieser, P. Frühwirt, P. Kieseberg et al., *Guess Who's Texting You? Evaluating the Security of Smartphone Messaging Applications*, NDSS Citeseer, 2012.
- [53] V. Jain, D. R. Sahu, and D. S. Tomar, "Evidence gathering of LINE messenger on iPhones," *International Journal For Innovative Engineering and Management*, vol. 4, no. 2, pp. 1–9, 2015.
- [54] I. Riadi, Sunardi, and A. Firdonsyah, "Forensic investigation technique on Android's BlackBerry Messenger using NIST framework," *International Journal of Cyber-Security and Digital Forensics*, vol. 6, no. 4, pp. 198–205, 2017.
- [55] M. K. Tri, I. Riadi, and Y. Prayudi, "Forensics acquisition and analysis ethod of IMO messenger," *International Journal of Computer Applications*, vol. 179, pp. 9–14, 2018.
- [56] A. Azfar, K. K. R. Choo, and L. Liu, "An Android social app forensics adversary model," in *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pp. 5597–5606, IEEE, 2016.
- [57] C. Wu, C. Vance, R. Boggs, and T. Fenger, *Forensic Analysis of Data Transience Applications in iOS and Android*, Marshall Forensic Science, 2013.
- [58] DB Browser for SQLite, *Sqlitebrowserorg* August 2019 <https://sqlitebrowser.org/>.
- [59] Tencent, *Tencentcom* August 2019 <https://www.tencent.com/en-us/>.
- [60] "SQLCIPHER-Zetetic," August 2019 <https://www.zetetic.net/sqlcipher/>.
- [61] ppwwyyxx, "wechat-dump," 2018, September 2019 <https://github.com/ppwwyyxx/wechat-dump>.
- [62] F. Darus, *How to Decrypt WeChat EnMicroMsg.db Database? Forensic Focus*, 2014, <https://www.forensicfocus.com/articles/decrypt-wechat-enmicromsgdb-database/>.
- [63] S. Wu, Y. Zhang, X. Wang, X. Xiong, and L. Du, "Forensic analysis of WeChat on Android smartphones," *Digital Investigation*, vol. 21, pp. 3–10, 2017.
- [64] Z. Dai, Sufatrio, T. W. Chua, D. K. Balakrishnan, and V. L. Thing, *Chat-App Decryption Key Extraction through Information Flow Analysis*, IOS Press, 2017.
- [65] H. Md and M. Warnier, *Cyber Crime in Privately Held Information Systems: Personal Data at Stake*, IEEE, 2014.
- [66] Z. Whittaker, *Wickr Technical White Paper*, 2019, September 2019 <https://www.documentcloud.org/documents/3461863-Wickrtechnical-white-paper.html>.

Research Article

Reducing Entropy Overestimation in Soft Actor Critic Using Dual Policy Network

Hamid Ali ¹, Hammad Majeed ¹, Imran Usman ², and Khaled A. Almejalli ²

¹National University of Computer and Emerging Sciences, Pakistan

²College of Computing and Informatics, Saudi Electronic University, Riyadh, Saudi Arabia

Correspondence should be addressed to Hammad Majeed; research.submission@gmail.com

Received 18 March 2021; Revised 11 May 2021; Accepted 29 May 2021; Published 10 June 2021

Academic Editor: Haider Abbas

Copyright © 2021 Hamid Ali et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In reinforcement learning (RL), an agent learns an environment through hit and trail. This behavior allows the agent to learn in complex and difficult environments. In RL, the agent normally learns the given environment by exploring or exploiting. Most of the algorithms suffer from under exploration in the latter stage of the episodes. Recently, an off-policy algorithm called soft actor critic (SAC) is proposed that overcomes this problem by maximizing entropy as it learns the environment. In it, the agent tries to maximize entropy along with the expected discounted rewards. In SAC, the agent tries to be as random as possible while moving towards the maximum reward. This randomness allows the agent to explore the environment and stops it from getting stuck into local optima. We believe that maximizing the entropy causes the overestimation of entropy term which results in slow policy learning. This is because of the drastic change in action distribution whenever agent revisits the similar states. To overcome this problem, we propose a dual policy optimization framework, in which two independent policies are trained. Both the policies try to maximize entropy by choosing actions against the minimum entropy to reduce the overestimation. The use of two policies result in better and faster convergence. We demonstrate our approach on different well known continuous control simulated environments. Results show that our proposed technique achieves better results against state of the art SAC algorithm and learns better policies.

1. Introduction

The agent with the ability of decision-making learns by interacting with the environment. First, it interacts and then explores the environment to learn different behaviors. This process is slow and often takes a lot of time to fully understand the dynamics of the environment. Based on an agent's ability, the learning time may increase with the increase in difficulty level of the environment.

Reinforcement learning (RL) is a branch of machine learning in which agent's goal is to learn a policy that enables it to take sequence of decisions while maximizing expected discounted sum of rewards. RL combined with deep learning enables RL agent to learn difficult task such as playing a game of Go [1], learning to play games from raw pixels [2] (playing Atari games), or performing continuous control tasks [3] (controlling humanoids in simulations). Use of neural networks as function approximates has removed the need of

manual feature engineering, where policies are directly optimized from raw pixels or sensors output. However, use of neural networks poses many challenges. Two major challenges are that (1) they need high number of samples for learning because neural networks are slow learners. This makes NN infeasible to use in real world problems. (2) Their performance is very sensitive to hyperparameters and require significant amount of time to search for good hyperparameters. Bad choice of parameters can degrade the performance of algorithm and can cause unstable learning. Because of these problems, RL methods cannot be successfully applied to many real-life problems. Our focus in this paper is on continuous control problems and how we can guide our agent solving these problems to train as fast as possible.

One of the major reasons for high sampling complexity is the use of on-policy learning. In on-policy, the policy interacting with the environment is the same as that is being

trained. This means new samples are required for every training step. Recently proposed methods such as TRPO [4] and PPO [5] allow multiple gradient updates on same samples and require less samples as compared to other on-policy methods. But still, they are very data hungry. On the other hand, off-policy methods are less data hungry and can reuse past experiences. Sarsa max [6] and DQN [7] are good examples of off-policy learning. Off-policy learning combined with function approximations such as neural networks are unstable and often diverge. This creates further challenges when action space is continuous. A classical example of this architecture is Deep Deterministic Policy Gradient (DDPG) [8]. In case of high-dimensional states and actions, DDPG suffers and gets stuck into local optima. It creates high peak for some actions and then stops the exploration. To overcome this problem, soft actor critics (SAC) [9, 10] is introduced which adds an entropy term to the objective function. In SAC, the goal is to maximize expected rewards as well as the entropy term. This entropy causes uncertainty and stops agent from creating high peaks for some actions. This results in better exploration and fast learning. In this paper, we have studied the effects of maximizing and minimizing the entropy on the learning process. We have found that maximizing entropy causes an overestimation of the entropy term, which results in unstable learning and slow convergence. We propose that by using dual policy, the effects of overestimation can be reduced which can result in stable, better, and faster learning. Furthermore, it helps in achieving better sampling efficiency.

Exploration means looking for new knowledge, and exploitation means tweaking existing knowledge in search for optimal policy. For better learning, it is very important for an agent to perform both to understand how environment works. Most RL algorithms suffer because they are unable to explore the environment in a structured way. This problem further increases when we work with continuous action space [11], where there is a high chance of agent getting stuck in local optima. In value-based methods, exploration can be achieved through explicit interaction with the environment. Best way to explore environment in value-based methods is through GILE [7] algorithm. At the start, it completely explores the environment and then starts the exploitation.

In policy-based methods, exploration is achieved through adding entropy to the policy [2, 4]. This approach fails when we have a large action space [7]. SAC [9, 10] also works with the same idea but it adds entropy to the objective function and maximizes it during the learning process. This helps the agent to explore the environment especially in case to continuous and large action space. As the agent learns, the entropy term is optimized which causes better exploration. Our focus in this paper is on the effects of entropy on the learning process in continuous action space. Our aim is to improve entropy estimate so that policy can be guided towards the optimal path as fast as possible.

Most existing RL algorithms work with deterministic policies. This is because deterministic policies are easy to optimize and result in stable convergence [12–14]. However deterministic policies are not good for exploration [15]. For

exploration, stochastic policies are used which return probability distribution of actions. Normally, exploration in stochastic policy is achieved through heuristics. By heuristics, we mean adding entropy to the stochastic policy or adding random noise. Stochastic policy helps the agent to explore environment in a structured way. Because of stochastic behavior, the agent can better generalize against the unseen states. In SAC, entropy is added to the objective function, where it is maximized along with the expected reward. We have seen that adding large entropy causes instability or slow convergence.

Actor critic methods are at the intersection of value-based methods such as DQN [7] and policy based such as reinforce [16]. Actor acts as a policy whose job is to output an optimal action. Critic’s job is to evaluate actor’s output using current state and action. Figure 1 shows a general actor critic framework. Actor is optimized using policy gradient method, and critic is optimized using value-based methods. Actor output is used to train critic, and then, critic is used to guide policy towards the optimal path.

Actor critic algorithms start learning from policy iteration. Policy iteration alternates between policy evaluation which means iteratively computing the value function for a policy and policy improvement which uses the value function output to iteratively obtain an optimal policy. For complex RL problems, it is not feasible to run both policy evaluation and policy improvement for infinite number of steps. Therefore, both are optimized concurrently. Basic actor critic framework uses entropy to encourage policy for exploration. Main difference between actor critic and SAC is that SAC adds entropy to the objective function (1).

$$\pi^* = \arg \max_{\pi} \sum_t \mathbb{E}_{(s_t, \mathbf{a}_t) \sim \rho_{\pi}} [r(s_t, \mathbf{a}_t) + \alpha \mathcal{H}(\pi(\cdot | s_t))]. \quad (1)$$

SAC maximizes entropy to encourage exploration. This maximization results in the overestimation of entropy. It is observed that the maximizing entropy can cause instability, especially in the early stages of learning when critic is not trained to differentiate between good and bad states. Normally, alpha temperature is used to reduce the effects of entropy term. Recently proposed SAC [10] uses entropy to optimally derive the value of alpha. That means alpha itself is dependent on entropy term. Maximizing entropy results in high alpha which means further adding uncertainty. To reduce this overestimation bias, we propose to train two different policies. During training iterations, we select the policy which gives us low entropy. Here, objective function is not changed, which is maximizing the entropy. To reduce the overestimation bias, we are choosing the policy which minimizes the entropy. This way, we are exploring the environment in structured way while making sure we are not overestimating the benefits of exploration against returns.

We have tested our proposed approach on different continuous control simulated environments created by MuJoCo [17]. Results show that our approach achieves good sampling efficiency as compared to state of the art RL algorithms. Our approach is also easy to implement and can result in better and faster learning.

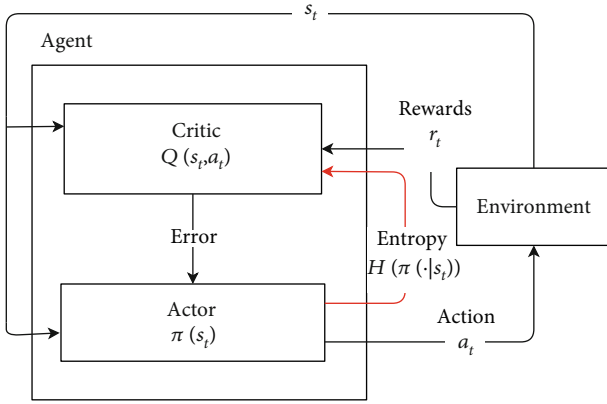


FIGURE 1: Black path shows actor critic framework. red path shows soft actor critic framework with entropy term.

2. Literature Review

In this section, we will first discuss some of the value-based methods which suffer from overestimation of action values and then different techniques to solve this problem. Then, we will discuss some policy gradient methods which work with continuous action spaces. Thirdly, we are going to discuss some state of the art methods which work with continuous actions. Lastly, we shall discuss the reasons of better performance of SAC and how SAC suffers from overestimation of entropy term.

2.1. Value-Based Methods. In this section, we are going to discuss value-based methods and how they achieve good sample efficiency and reduce overestimation in entropy-based methods.

2.1.1. Deep Q Learning. Deep Q-Network (DQN) [4] combines neural networks with Q learning, where images (continuous state spaces) are passed as input to deep neural network. NN represents an action value function where last layer of NN corresponds to number of actions agent can take, and optimal policy is derived once optimal value function is found. Because DQN outputs the optimal action, so it cannot be used for continuous actions. DQN provides high sampling efficiency using experience replay. Experience replay is used to break the correlation between consecutive tuples that agent sees while encountering with environment. From deep learning, we have seen that NN overfits if same data is passed in same order again and again. Also, we do not want our agent to learn a sequence of state action pairs; rather, we want agent to learn a best policy. By randomly picking, data from experience replay helps us to break this correlation between sequence to state action pairs. SAC uses the same experience replay to achieve high sampling efficiency.

Q learning uses one step bootstrapping, and at every time step, it tries to minimize mean-squared error between the current state and next state.

$$L_i(\theta_i) = \mathbb{E}_{(s,a,r,s') \sim U(D)} \left[\left(r + \gamma \max_{a'} Q(s', a'; \theta_i) - Q(s, a; \theta_i) \right)^2 \right]. \quad (2)$$

Values of NN are evolved at each time step t by using (2). It creates a problem while converging because we are chasing a moving target. At every time step, when we move a head, our target moves too. This results in nonconvergence of NN. To overcome this problem, DQN uses fixed Q target or target network where a separate NN is used to find next state values. Values of policy are replaced with target network after 1000 number of interactions.

Q learning uses greedy policy to learn the environment which means to calculate the value of next state, it uses max value action in the next state using (2). This maximization results in overestimation of action values and creates high variance, especially when we are at the early stage of learning because Q values evolve very fast. Following approaches were proposed to reduce this overestimation.

2.1.2. Double Q Learning. First solution to address overestimation of action values was proposed in [18]. This paper proposed to use two different Q value estimators where action was chosen from both the estimators. Then, with equal probability, Q estimators are updated. At each time step, only one Q value estimator is updated. By using two different estimators, overestimation was reduced which resulted in reducing variance in Q values.

2.1.3. Double Deep Q Learning. Another solution to address the problem of overestimation of action values was proposed in double Q learning [19]. To overcome the problem of moving target values, a separate network was maintained which is known as target network in DQN [7]. In double DQN, the authors proposed the idea that instead of maintaining two separate estimators like in [18], they selected action using model network and got its Q value using target network. From (3), it is clear that model network was used to select action with maximum value but to obtain final Q value target network was used. In (3), θ_t represents model weights, and θ'_t represents target network weights.

$$Y_t^{\text{DoubleQ}} \equiv R_{t+1} + \gamma Q \left(S_{t+1}, \underset{a}{\operatorname{argmax}} Q(S_{t+1}, a; \theta_t); \theta'_t \right). \quad (3)$$

2.2. Policy Gradient Methods. Now, we are going to discuss policy gradient methods and how they use Monte Carlo estimate to update the policy. Policy gradient methods are on-policy methods. Gradient ascent policy tries to maximize expected return while directly searching for optimal policy. Policy gradient methods work with trajectories where trajectory is just an episode with a fixed number of interaction with the environment. Policy gradient methods calculate Monte Carlo estimate over trajectories.

2.2.1. Reinforce. Reinforce [13] is a policy gradient algorithm where gradient of expected return is only calculated with respect to current policy parameters. Gradient of a trajectory is calculated with respect to policy parameters. For this, trajectory must be representative of the current policy. This makes reinforce very data hungry because it learns once from the collected data. Reinforce also suffers from credit assignment problem because Monte Carlo calculates the sum of

rewards at each time step. Here, Markov assumption is used which says to predict the future, you only need present.

2.2.2. Trust Region Policy Optimization. Trust Region Policy Optimization (TRPO) [4] uses the concept of Important Sampling (IS) for making the RL less data hungry. TRPO uses trajectories of the old policy to update current policy. This is achieved through a surrogate function (5). Surrogate function is used to compute the gradient of old policy. Furthermore, it enables us to use same batch of data multiple times to update policy. Here, π_{θ_t} is the current policy which is updated through old policy $\pi_{\theta_{\text{old}}}$. \hat{A}_t is an advantage function.

$$\text{maximize}_{\theta} \hat{\mathbb{E}}_t \left[\frac{\pi_{\theta}(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)} \hat{A}_t \right], \quad (4)$$

$$\text{subject to } \hat{\mathbb{E}}_t [\text{KL}[\pi_{\theta_{\text{old}}}(\cdot|s_t), \pi_{\theta}(\cdot|s_t)]] \leq \delta. \quad (5)$$

In theory, IS helps us to use old trajectories as many times as we want. But in practice, TRPO suffers from high variance and often diverge to unrecoverable path. Here, trust region methods are proposed to reduce variance. TRPO establishes a trust region using KL divergence (6) to prevent current policy from deviating too far from old policy. In (6), β is some coefficient which is used to enforce hard constraint.

$$\text{maximize}_{\theta} \hat{\mathbb{E}}_t \left[\frac{\pi_{\theta}(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)} \hat{A}_t - \beta \text{KL}[\pi_{\theta_{\text{old}}}(\cdot|s_t), \pi_{\theta}(\cdot|s_t)] \right]. \quad (6)$$

TRPO can work with both discrete and continuous action spaces. TRPO adds entropy in policy for exploration which is hindered by the surrogate function. Recall that surrogate function stops the agent from going too far from the old policy. This means new policy cannot explore environment independently.

2.2.3. Proximal Policy Optimization. Proximal policy optimization (PPO) [5] is improved version of TRPO. In PPO, instead of establishing trust region, a policy clipping method (7) is used, clipping a policy between minimum of $1 - \epsilon$, $1 + \epsilon$. This makes sure that new policy does not diverge more than ϵ distance away from old policy. PPO clipping function is very important because it reduces variance and makes sure that old and new policies are not too far away from each other.

$$L^{\text{CLIP}}(\theta) = \hat{\mathbb{E}}_t [\min(w_t(\theta)\hat{A}_t, \text{clip}(w_t(\theta), 1 - \epsilon, 1 + \epsilon)\hat{A}_t)]. \quad (7)$$

$w_t(\theta)$ will be greater than one if the particular action is more probable for the current policy than it is for the old policy. It will be between 0 and 1 when the action is less probable for our current policy. \hat{A}_t is an advantage function.

PPO initially allows policy to explore but with time, as the number of interactions increases, it gets stuck in local optima because policy is clipped if it moves too far away from old

policy. Due to high sampling cost, it is not feasible to apply PPO on real world problems.

2.2.4. A3C and A2C. Both DQN and reinforce either suffer from variance or biasness. To create a balance between both estimates Asynchronous Advantage Actor Critic (A3C) [2] was proposed which uses N step bootstrapping. Actor learns advantage function and critic learns value function. Instead of using experience replay to break correlation between consecutive tuples, A3C creates multiple agents of environment at the same time. Every agent interacts with environment for n number of times to collect experiences. At any time step, the agent will receive multiple minibatches of correlated experiences. Experience will be decor-related because agent will see multiple states and actions at the same time. Asynchronous here means that every agent will have its own copy of environment, and it can be different from what other agents have. Different approach called synchronous actor critic (A2C) [3] uses same copy of environment for every agent. In A2C, all agents wait to complete segment of interaction, then network is updated once for all agents. A2C gives equal or better results than A3C.

A3C and A2C are faster as compared to TRPO and PPO because they create multiple copies of the same environment and can learn parallel. This is useful in case of simulated environments but cannot be applied to real world. Also, creating multiple copies and training them in parallel only helps with a faster learning. It does not help agent in terms of sampling efficiency.

2.3. Continuous Action Space Methods. Now, we are going to look some of the actor critic methods that can only work with continuous action spaces. These methods achieve good sampling efficiency as compared to on-policy methods because they uses off-policy updates. Also, these methods can work with large action spaces where on-policy methods suffer from curse of dimensionality of action space.

2.3.1. Deep Deterministic Policy Gradients (DDPG). DDPG was proposed in [8]. DDPG is a combination of both value- and policy-based methods which searches for deterministic policy and works with only continuous actions. DDPG has four neural networks, a deterministic policy network θ^{μ} , Q network θ^Q , a target policy network $\theta^{\mu'}$, and target Q network $\theta^{Q'}$. DDPG directly maps states into action spaces. Target networks are the delayed copies of original networks to increase stability. Like DQN [7], DDPG also uses experience replay with off-policy updates to achieve high sample efficiency. In DDPG, the value network loss is calculated using Bellman equation (8). Then, value networks are updated using the mean-squared error between current and next Q values. SAC also use the same Bellman equation to update value networks. The only difference is that SAC adds entropy with next Q value to stop Q networks from creating high peaks for some actions.

$$y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1} | \theta^{\mu'}) | \theta^{Q'}). \quad (8)$$

To update policy, DDPG takes the derivative of expected return with respect to policy parameters θ . Then, it takes mean of gradients against minibatch that is sampled from experience replay to calculate policy loss.

$$J(\theta) = \mathbb{E} \left[Q(s, a) \Big|_{s=s_t, a_t=\mu(s_t)} \right]. \quad (9)$$

To encourage policy for exploration, DDPG adds time-correlated noise [8]. But recent research says that uncorrelated, mean-zero Gaussian noise [10] also works. In SAC, exploration is achieved by adding entropy with policy (13).

$$\mu'(s_t) = \mu(s_t | \theta_t^\mu) + \mathcal{N}. \quad (10)$$

Instead of replacing target networks after a fixed number of times, DDPG uses soft target updates in which target network weights are slowly updated by original networks weights. SAC also use soft target updates to update target networks.

2.3.2. Twin-Delayed DDPG (TD3). In [20], a new algorithm called Twin-Delayed Deep Deterministic policy gradients (TD3) was proposed. In TD3, old idea of double Q learning [18] is used. Twin in TD3 means that we are using two networks for critic. TD3 uses clipped Q learning where the minimum value of critic is used. This results in the underestimation of Q values thus better and stable learning curve. Our idea of reducing overestimation from entropy is inspired from TD3. SAC also uses the same approach to reduce overestimation in action values. Like TD3, we have decided to use two different policies to reduce overestimation in entropy term.

To encourage exploration, TD3 uses target policy smoothing. In DDPG during training, high peaks are created for some actions. These high peaks push policy towards picking action and stop the policy from exploration. This can be minimized by adding random noise (ϵ) [11] which act as a regularization. Noise is clipped to make sure it lies with in valid action range.

$$a'(s') = \text{clip} \left(\mu_{\theta_{\text{arg}}}(s') + \text{clip}(\epsilon, -c, c), a_{\text{Low}}, a_{\text{High}} \right), \quad \epsilon \sim \mathcal{N}(0, \sigma). \quad (11)$$

3. Soft Actor Critic (SAC)

Generally, in RL, the goal is to learn a deterministic or stochastic policy to maximize the expected sum of rewards by using (12).

$$\sum_t \mathbb{E}_{(s_t, \mathbf{a}_t) \sim \rho_\pi} [r(s_t, \mathbf{a}_t)]. \quad (12)$$

Standard RL algorithms trap in local optima due to reliance on only the reward (overestimation). Therefore, whenever a high peak in performance is achieved in some trajectory, it would be almost impossible to recover from there. Where as in SAC, the goal is to learn a stochastic policy

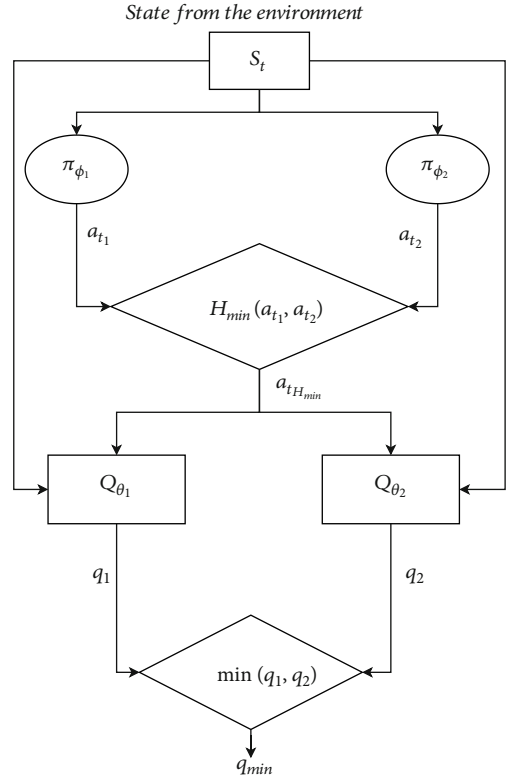


FIGURE 2: Proposed dual-policy soft actor critic. State s_t is passed on to both policies π_{θ_1} and π_{θ_2} . Entropy \mathcal{H} against the actions a_{t_1} and a_{t_2} generated by the policies π_{θ_1} and π_{θ_2} are calculated. Finally, the action against minimum entropy is selected to reduce overestimation. The rest of the process is same as it was in soft actor critic [9].

that maximizes both the expected sum of rewards and entropy by using (13).

$$J(\pi) = \sum_t \mathbb{E}_{(s_t, \mathbf{a}_t) \sim \rho_\pi} [r(s_t, \mathbf{a}_t) + \alpha \mathcal{H}(\pi(\cdot | s_t))], \quad (13)$$

where ρ_π is the state marginals of the trajectory induced by a policy $\pi(\cdot | s_t)$, α is the temperature, and the rest of the terms have conventional meanings. α determines the relative importance of the reward over entropy. This effectively controls the stochasticity of the optimal policy. By setting $\alpha = 0$, SAC can be transformed into a standard RL that only maximizes the sum of rewards. Above-mentioned objective term (13) has many advantages. For example, it greatly rewards the policy with the ability of high exploration and rejecting the unpromising areas of the search space. Furthermore, it gives equal opportunity of selection to nearly equal good actions. This resolves the problem of overestimation in conventional RL to some extent. SAC can be extended to apply on infinite horizon problems by introducing a discount factor γ . This ensures that the sum of the expected rewards and entropy is finite. Figure 1 shows the graphical representation of SAC framework. Two versions of SAC are proposed. They are discussed in detail in the following sections.

```

1: Set initial policy parameters  $\phi_1, \phi_2$ , Q-function parameters  $\theta_1, \theta_2$ , empty replay buffer  $\mathcal{D}$ ,  $\gamma$  discount reward
2: Set target parameters equal to main parameters  $\theta_{tar_{g_1}} \leftarrow \theta_1, \theta_{tar_{g_2}} \leftarrow \theta_2$ 
3: repeat
4: {Observe state  $s_t$  and select action against minimum entropy}
5:  $a_{t_k} \sim \pi_{\phi_k}(\cdot | s_t)$  for  $k = 1, 2$ 
6:  $e_m, m \leftarrow \min_{k=1,2} \log_{\pi_{\phi_k}}(a_{t_k} | s_t)$  { $m$  is the index of the policy with minimum entropy}
7: Execute  $a_{t_m}$  in the environment
8: Observe next state  $s'$ , reward  $r$ , and done signal  $d$  to indicate whether  $s'$  is terminal
9: Store  $(s, a_{t_m}, r, s', d)$  in replay buffer  $\mathcal{D}$ 
10: if  $s'$  is terminal
11:   reset environment state
12: end if
13: if it is time to update then
14:   for  $j$  in range(no. of updates required) do
15:     Randomly sample a batch of transitions,  $B = (s, a, r, s', d)$  from  $\mathcal{D}$ 
16:     Compute targets for the Q functions:
17:      $\tilde{a}'_k \sim \pi_{\phi_k}(\cdot | s')$  for  $k = 1, 2$ 
18:      $e'_m, m \leftarrow \min_{k=1,2} \log_{\pi_{\phi_k}}(\tilde{a}'_k | s')$  { $m$  is the index of the policy with minimum entropy}
19:      $y(r, s', d) = r + \gamma(1 - d)(\min_{i=1,2} Q_{\theta_{tar_{g_i}}}(s', \tilde{a}'_m) - \alpha e'_m)$ 
20:     Update Q-functions by one step of gradient descent using
21:      $\nabla_{\theta_i} (1/|B|) \sum_{(s,a,r,s',d) \in B} (Q_{\theta_i}(s, a) - y(r, s', d))^2$  for  $i = 1, 2$ 
22:     Update policy by one step of gradient ascent using
23:      $\tilde{a}_k \sim \pi_{\phi_k}(\cdot | s)$  for  $k = 1, 2$ 
24:      $e_m, m \leftarrow \min_{k=1,2} \log_{\pi_{\phi_k}}(\tilde{a}_k | s)$  { $m$  is the index of the policy with minimum entropy}
25:      $\nabla_{\phi_m} (1/|B|) \sum_{s \in B} \mathbb{E}(\min_{i=1,2} Q_{\theta_i}(s, \tilde{a}_m) - \alpha e_m)$ ,
26:     {where  $\tilde{a}_m$  is a sample from  $\pi_{\phi_m}$  which is differentiable w.r.t.  $\phi_m$  via the reparametrization trick.}
27:     Update target networks with:
28:      $\theta_{tar_{g_i}} \leftarrow \rho \theta_{tar_{g_i}} + (1 - \rho) \theta_i$  for  $i = 1, 2$ 
29:     {where  $\rho$  is polyak. (Always between 0 and 1, usually close to 1.)}
30:   end for
31: end if
32: until convergence

```

ALGORITHM 1: Proposed Soft Actor Critic Algorithm.

3.1. *Soft Actor Critic VI*. SAC was first proposed in [9]. In the first version, SAC uses fixed α . As discussed earlier, α is used to vary the importance of reward over entropy and vice versa in the objective function. First version of SAC comprises of a parameterized state value function $V_\psi(s_t)$, soft Q-function $Q_\theta(s_t, a_t)$, and a tractable policy $\pi_\phi(a_t | s_t)$. These functions are implemented as networks and use ψ , θ , and ϕ as parameters. A policy function parameterized with ϕ , a value function parameterized with ψ , and a Q function parameterized with θ . Q and value functions are related and help with stable convergence.

Value function is optimized using the objective function given in (14). It calculates the mean-squared error between the value network prediction and the prediction of the expected Q function plus the entropy term from the policy.

$$J_V(\psi) = \mathbb{E}_{s_t \sim \mathcal{D}} \left[\frac{1}{2} \left(V_\psi(s_t) - \mathbb{E}_{\mathbf{a}_t \sim \pi_\phi} [Q_\theta(s_t, \mathbf{a}_t) - \log \pi_\phi(\mathbf{a}_t | s_t)] \right)^2 \right], \quad (14)$$

where \mathcal{D} is a replay buffer.

Q network is optimized by minimizing the error expression of (15).

$$J_Q(\theta) = \mathbb{E}_{(s_t, a_t) \sim \mathcal{D}} \left[\frac{1}{2} (Q_\theta(s_t, a_t) - Q^\wedge(s_t, a_t))^2 \right], \quad (15)$$

where

$$\widehat{Q}(s_t, \mathbf{a}_t) = r(s_t, \mathbf{a}_t) + \gamma \mathbb{E}_{s_{t+1} \sim p} [V_{\tilde{\psi}}(s_{t+1})]. \quad (16)$$

During this optimization, for all the (state, action) pairs in the replay buffer, we want to minimize the squared difference between the prediction of our Q function and the immediate (one time-step) reward plus the discounted expected value of the next state. $V_{\tilde{\psi}}$ is the target value function which is updated after updating all the three networks.

Stochastic policy network outputs mean and standard deviation across all the actions. Actions are then sampled from the normal distribution which is created by using the mean and standard deviation generated by the policy. In

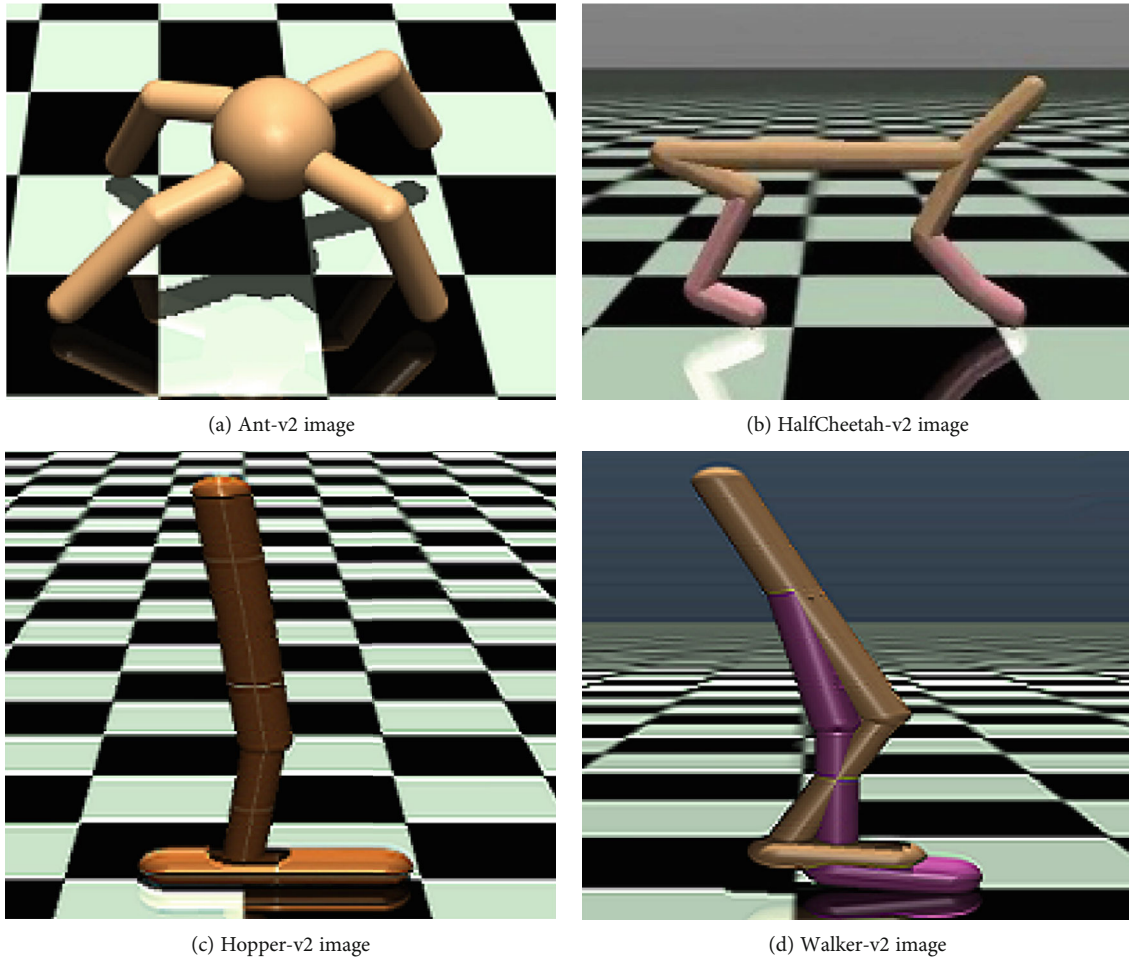


FIGURE 3: Four environments used in our experiments.

[9], reparameterization trick is introduced to ensure that policy function remains differentiable while applying backpropagation. The new policy is defined as (17).

$$a_t = f_\phi(\varepsilon_t; s_t), \quad (17)$$

where ε is a noise vector sampled from a Gaussian distribution (\mathcal{N}). By using this new objective function, policy network can be optimized by using (18).

$$J_\pi(\phi) = \mathbb{E}_{s_t \sim \mathcal{D}, \varepsilon_t \sim \mathcal{N}} \left[\log \pi_\phi \left(f_\phi(\varepsilon_t; s_t) | s_t \right) - Q_\theta \left(s_t, f_\phi(\varepsilon_t; s_t) \right) \right]. \quad (18)$$

3.2. Soft Actor Critic V2. The second version of SAC are proposed in [10]. Two things are changed in version 2. Value network is removed, and temperature value is automatically determined. The policy network is same as it was in version 1. According to [10],

“Simply forcing the entropy to a fixed value is a poor solution, since the policy should be free to explore more in regions where the optimal action is uncertain, but remain more deterministic in states with a clear distinction between good and bad actions.”

In SAC V2, α is optimized by using the objective function (19)

$$J_\alpha = \mathbb{E} \left[-\alpha \log \pi_t(a_t | s_t; \alpha) - \alpha \bar{\mathcal{H}} \right], \quad (19)$$

where $\bar{\mathcal{H}}$ is the desired entropy. In our research, we have found out that scaling down entropy with alpha still results in the overestimation of action values. It is because alpha parameter is derived from entropy term.

4. Proposed Methodology

In this section, we will discuss our proposed approach in detail. As discussed earlier, RL algorithms suffer from overestimation and the best way to address it by using two different neural networks [20]. Our technique of reducing overestimation in entropy is based on SAC and inspired from this technique. We propose to use two different policies π_{ϕ_1} and π_{ϕ_2} , parameterized by ϕ_1 and ϕ_2 . Both the policies are initialized randomly. As entropy is determined from the policy itself therefore, the use of different policies enable us to estimate different entropy values. Figure 2 shows work flow of our proposed soft actor critic. For example, let us assume state

TABLE 1: Hyperparameters.

Parameter	Value
Optimizer	Adam ([21])
Max time steps	1 million
Learning rate	$3 \cdot 10^{-4}$
Discount (γ)	0.99
Replay buffer size	10^6
# of hidden layers	2
# of hidden units per layer	256
# of samples per minibatch	256
Entropy target	$-\dim(\mathcal{A})$ (e.g., -8 for Ant-v2)
Nonlinearity	ReLU
Target smoothing coefficient (τ)	0.005
Target update interval	1
Gradient steps critic	1
Reward scaling	1.0
Initial alpha	0.2
Alpha learning rate	$3e-4$
Evaluation frequency	$5e3$
Evaluation episodes	10
Exploration time steps	$25e3$ for Ant-v2 and HalfCheetah $1e3$ for Hopper-v2 and Walker2d

s_t from the environment is fed to our agent. Both the policies π_{ϕ_1} and π_{ϕ_2} get the same state and output mean and standard deviation across all the actions used. Actions a_1 and a_2 are sampled from the normal distributions that are constructed from the means and standard deviations returned by the policies (20). Then, entropy is calculated against both the policies. The final action is selected against the minimum entropy using (21).

$$a_k \sim \pi_{\phi_k}(\cdot|s_t) \text{ for } k = 1, 2, \quad (20)$$

$$\min_{k=1,2} \log_{\pi_{\phi_k}}(a_k|s_t). \quad (21)$$

Note that in our proposed approach, we are not changing the objective function which is to maximize entropy along with expected reward. Entropy is added to encourage exploration so that agent does not get stuck in local optima by creating high peaks for some actions. By using two policies, we are encouraging our agent to maximize entropy where the action distribution is relatively normal. Entropy makes sure that it does not create high peaks by repeatedly selecting the same action. We are optimizing the policy against low entropy which helps agent to learn the optimal action in case of unsure situations. Optimizing different policies at the same time enables exploration which was previously dependent on only the entropy. It also helps agent to capture different versions of optimal policy and stops it from premature convergence to local optima.

In [9], it is shown that alpha is a sensitive hyperparameter. Wrong alpha can drastically degrade the performance

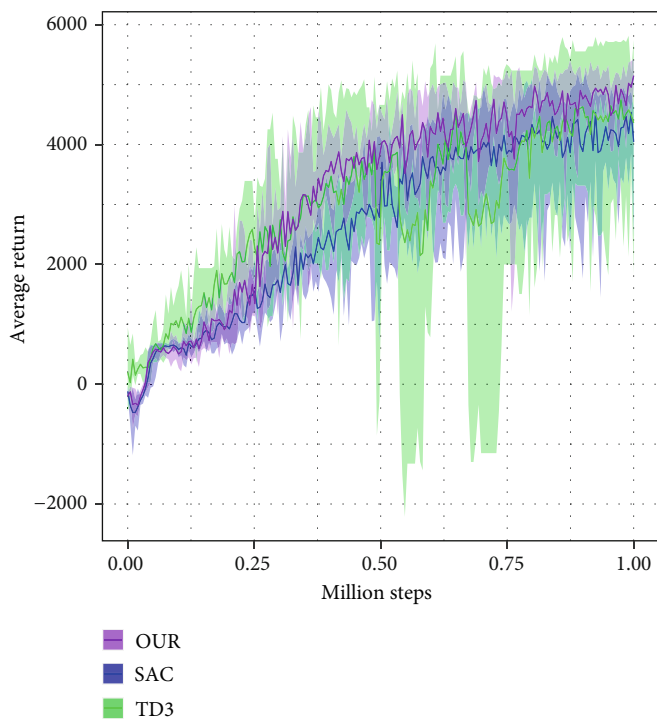
of our algorithm. Also, reward signal not only varies across different tasks but also changes within the same task as policy learns more about the environment. As the agent learns, entropy also changes with policy. Fixing policy to a static entropy value is a bad idea because it restricts the agent to explore the environment. Especially in regions where it is unclear about the optimal action. Complete algorithm of SAC with dual policy optimization is described in Algorithm 1.

In Algorithm 1, lines 1-9 are collecting and storing data in replay buffer for future learning. In line 6, e_m represents the minimum entropy produced by our dual policies. m contains the index of the policy generating e_m . Lines 13-30 perform the update of Q and π networks. In lines 17-18 and 23-24, minimum entropy (e_m) along with the index of the policy generating the minimum entropy (m) are recorded. These are then used to update the Q and π networks.

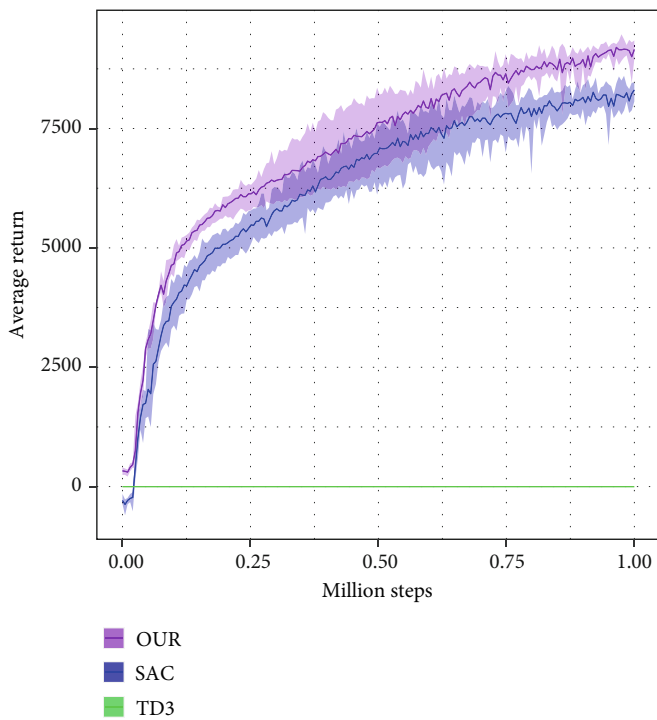
In pure stochastic environment, both the policies (π_{ϕ_1} and π_{ϕ_2}) will be updated with equal probability. We believe that the delayed update of the policies has twofold benefits: (1) it stops one policy to take over by selecting some range of actions (overestimation avoidance); 2) results in quality update to the policy. Therefore, the policies will converge to optimal quickly.

5. Experimental Setup

We have tested our approach on different simulated environments created by MuJoCo [17]. MuJoCo is a library for modeling, simulation and visualization of multijoint

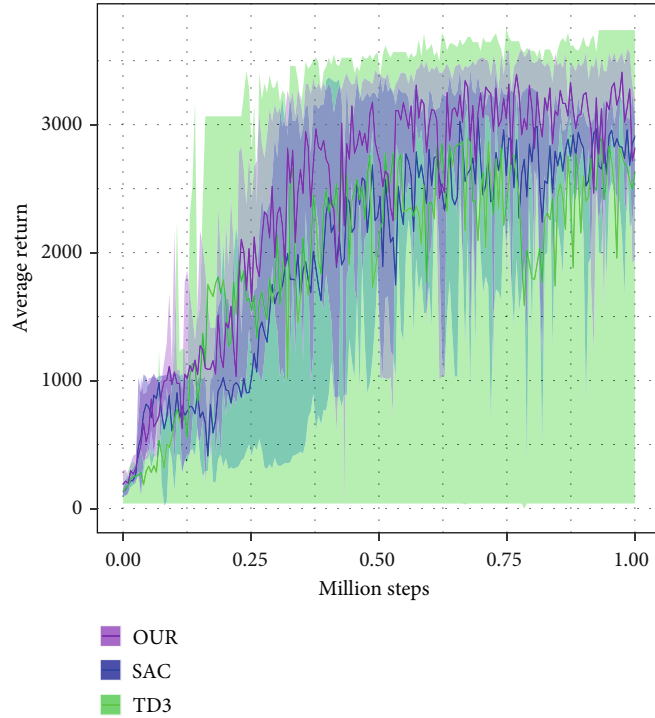


(a) Ant-v2 results

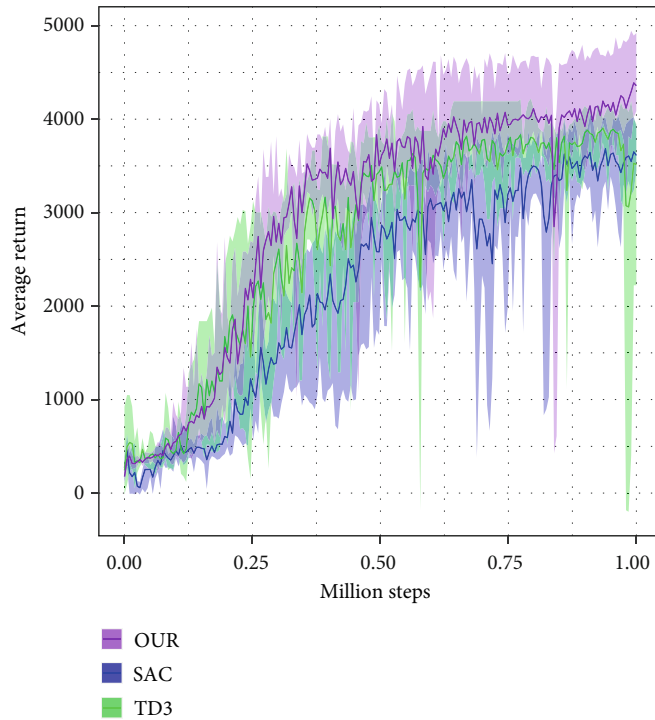


(b) HalfCheetah-v2 results

FIGURE 4: Continued.



(c) Hopper-v2 results



(d) Walker-v2 results

FIGURE 4: Results of different setups on the four MuJoCo environments [17]. Training curves of SAC (blue), TD3 (green), and our proposed dual policy SAC (purple) are shown.

dynamics with contact. Both state and action spaces are continuous for all the environments. In our experiments, reward range is not bounded in specific range. Furthermore, we did not scale reward for any environment.

5.1. Environments Used. Figure 3 shows the environments that are used in our experiments.

First environment is Ant-v2 (look at Figure 3(a)). In Ant-v2, the goal is to make a four-legged ant walk as fast as

TABLE 2: Max evaluation reward achieved during learning.

Environment	Algorithm	Max value	S.D
Ant-v2	SAC	4536.18	1425.31
	TD3	4360.79	2081.05
	Ours	5151.27	1600.29
HalfCheetah-v2	SAC	8308.59	4298.95
	TD3	-1.65	0
	Ours	9158.21	4414.13
Hopper-v2	SAC	2905.15	1388.34
	TD3	2622.66	1245.89
	Ours	2812.76	1311.20
Walker2d-v2	SAC	3611.99	1670.66
	TD3	3513.84	1635.82
	Ours	4357.33	2090.74

possible. It has 8 action dimensions and 111 state dimensions. All actions have range of -1 to +1. Environment reward range is infinite. Max time step to interact with the environment in one episode is 1000.

Second environment is HalfCheetah-v2 (look at Figure 3(b)). In HalfCheetah-v2, the goal is to make a two-legged cheetah walk as fast as possible. It has 6 action dimensions and 17 state dimensions. All actions have range of -1 to +1. Environment reward range is infinite. Max time step to interact with the environment in one episode is 1000.

Third environment is Hopper-v2 (look at Figure 3(c)). In Hopper-v2, the goal is to make a two-dimensional one-legged robot hop forward as fast as possible. It has 3 action dimensions and 11 state dimensions. All actions have range of -1 to +1. Environment reward range is infinite. Max time step to interact with the environment in one episode is 1000.

Fourth environment is Walker2d-v2 (look at Figure 3(d)). In Walker2d-v2, the goal is to make a two-legged bipedal robot walk as fast as possible. It has 6 action dimensions and 17 state dimensions. All actions have range of -1 to +1. Environment reward range is infinite. Max time step to interact with the environment in one episode is 1000.

We have defined the above-mentioned environments using Markov Decision Processes (MDP). An MDP consist of tuple (S, A, p, r) , where S is a set of state space, A is a set of action space, r is a set of rewards, and p is a state transition probability $p(s', r | s, a)$ also known as one step dynamics. State transition determines the probability of next state s' given current state $s_t \in S$ and action $a_t \in A$. Reward range should be finite such that $r : S \times A \rightarrow [r_{\min}, r_{\max}]$. We used $\rho_\pi(\tau)$ to represent trajectories $\tau = (s_0, a_0, s_1, a_1, \dots)$ gathered by a policy $\pi(a_t | s_t)$. Our focus is on environments with continuous state and action spaces. The goal of experimentation is to determine how dual policy SAC help us to achieve better sample efficiency than previous SAC. We have evaluated our technique on continuous control tasks from the OpenAI gym environments (toolkit for developing and comparing reinforcement learning algorithms). For our implementation of dual policy SAC, we have used two feed forward neural net-

works both with 256 neurons. On both actor and critic, we have used rectified linear units (ReLU) as an activation function. Critic receives both state and action as an input. Both networks are optimized using Adam optimizer [21] with a learning rate of $3e-4$. At each time step, both networks are trained for a minibatch of 264 samples. Samples are obtained uniformly from a replay buffer. Replay buffer has a length of 1 million samples. Check Table 1 for hyperparameter details. Table 1 shows the list of hyperparameter that are used for training of our proposed dual policy soft actor critic. To stop policy from getting stuck into the local optima, we have used exploratory policy at the start of learning. After that, interaction with environment was dependent on policy.

6. Results and Discussion

Since RL algorithm shows a lot of variance during learning therefore, we have calculated the average of the evaluation roll outs calculated during training. For better comparison, we have trained five different agents of same algorithm with random seeds (0, 1, 2, 3, 4). Each agent is trained for one million time steps. Each agent is evaluated after 5000 time steps for 10 episodes, and average of 10 episodes is recorded.

Results of the experiments are shown in Figure 4. The solid curves show the average value across all the five agents. Shaded region shows the minimum and maximum values at the evaluation time step. The results show that dual policy SAC outperforms existing state of art and achieved high sampling efficiency. Dual policy SAC also outperforms existing algorithms in terms of learning speed.

As agent's dynamics change with environment therefore, it is not always possible for the agent to learn on every environment [12, 22]. For this reason, TD3 is not able to train on HalfCheetah-V2 9. Our proposed dual-policy actor critic outperforms SAC while TD3 is not able to learn on this environment. It can be seen that our proposed dual-policy actor critic outperforms state of the art methods.

Table 2 shows the maximum reward that is achieved during the complete learning cycle and the average maximum reward that is achieved during evaluation roll-outs. It can be seen that other than Hopper-v2, our proposed dual-policy soft actor critic has achieved highest reward in all environments. Even in Hopper-v2, the highest reward of SAC is closed to what is achieved by our proposed algorithm. Last column shows the standard deviation.

7. Conclusion and Future Work

In this research, we have studied the difference between the objectives of SAC objective from the objective of conventional reinforcement learning algorithm. We have shown how maximum entropy framework works with RL objective and helps agent to explore environment in a structured way. We have discovered that maximizing the entropy results in the overestimation of entropy term. Recent proposed SAC automatically optimizes alpha using the entropy. This automation removes the need of manual alpha tuning but does not stop overestimation. It results in unstable and slow learning due to agent's divergence from the optimal path.

In this research, we have proposed a new dual policy optimization technique to reduce the effects of entropy overestimation. Optimizing two policies gives us two additional benefits. First, it helps agent to further explore the state and action space. Second, it helps agent to capture different modes of optimal policy. We have tested our approach on different continuous control simulated environments created by MuJoCo [17]. Results show that our approach has achieved good sample efficiency as compared to state of the art RL algorithms. Our approach is also easy to implement and can result in better and faster learning.

In the future, we can train agent to learn from prioritized samples. Prioritizing sample from experience replay is proposed in [23]. Instead of prioritizing samples based on Td error, we can prioritize samples based on entropy. To encourage exploration and stable learning, we can prioritize samples based on low entropy which can further push agent towards exploration.

Data Availability

We have used MuJoCo (advanced physics simulation) for our experiments. For using it, access to the MuJoCo library is required which is available on request from <https://www.roboti.us/license.html>. The details of the setup and the parameters used by us are given in experimental setup section of the paper.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] Z. Wang, T. Schaul, M. Hessel, H. Hasselt, M. Lanctot, and N. Freitas, "Dueling network architectures for deep reinforcement learning," in *33rd International Conference on Machine Learning, ICML 2016, vol. 4*, pp. 2939–2947, New York, USA, 2016.
- [2] V. Mnih, A. P. Badia, M. Mirza et al., "Asynchronous methods for deep reinforcement learning," in *33rd International Conference on Machine Learning, ICML 2016, vol. 4*, pp. 2850–2869, New York, USA, 2016.
- [3] M. Babaeizadeh, I. Frosio, S. Tyree, J. Clemons, and J. Kautz, "Reinforcement learning through asynchronous advantage actor-critic on a GPU," in *5th International Conference on Learning Representations, ICLR 2017- Conference Track Proceedings*, Palais des Congrès Neptune, Toulon, France, 2017.
- [4] J. Schulman, S. Levine, P. Moritz, M. Jordan, and P. Abbeel, "Trust region policy optimization," in *32nd International Conference on Machine Learning, ICML 2015, vol. 3*, pp. 1889–1897, Lille, France, 2015.
- [5] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, <http://arxiv.org/abs/1707.06347>.
- [6] S. Singh, T. Jaakkola, M. L. Littman, and C. Szepesvári, "Convergence results for single-step on-policy reinforcement-learning algorithms," *Machine Learning*, vol. 38, no. 3, pp. 287–308, 2000.
- [7] V. Mnih, K. Kavukcuoglu, D. Silver et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
- [8] T. P. Lillicrap, J. J. Hunt, A. Pritzel et al., "Continuous control with deep reinforcement learning," in *4th International Conference on Learning Representations, ICLR 2016- Conference Track Proceedings*, Caribe Hilton, San Juan, Puerto Rico, 2016.
- [9] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *35th International Conference on Machine Learning, ICML 2018, vol. 5*, pp. 2976–2989, January 2018, <http://arxiv.org/abs/1801.01290>.
- [10] T. Haarnoja, A. Zhou, K. Hartikainen et al., "Soft actor-critic algorithms and applications," 2018, <http://arxiv.org/abs/1812.05905>.
- [11] J. Schulman, P. Moritz, S. Levine, M. I. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," in *4th International Conference on Learning Representations, ICLR 2016- Conference Track Proceedings*, Caribe Hilton, San Juan, Puerto Rico, 2016.
- [12] J. Duan, G. Yang, S. E. Li, Y. Ren, and B. Cheng, "Distributional soft actor-critic: off-policy reinforcement learning for addressing value estimation errors," 2020, <http://arxiv.org/abs/2001.02811>.
- [13] X. Ma, L. Xia, Z. Zhou, J. Yang, and Q. Zhao, "DSAC: distributional soft actor critic for risk-sensitive reinforcement learning," 2020, <http://arxiv.org/abs/2004.14547>.
- [14] W. Zhou, Y. Li, Y. Yang, H. Wang, and T. M. Hospedales, "Online meta-critic learning for off-policy actor-critic methods," 2020, <http://arxiv.org/abs/2003.05334>.
- [15] M. Hessel, J. Modayil, H. Van Hasselt et al., "Rainbow: combining improvements in deep reinforcement learning," in *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pp. 3215–3222, Hilton New Orleans Riverside, New Orleans, Louisiana, USA, 2018.
- [16] P. S. Thomas and E. Brunskill, "Policy gradient methods for reinforcement learning with function approximation and action-dependent baselines," pp. 1057–1063, 2017, <http://arxiv.org/abs/1706.06643>.
- [17] E. Todorov, T. Erez, and Y. Tassa, "MuJoCo: a physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, Vila-moura-Algarve, Portugal, October 2012.
- [18] H. Van Hasselt, "Double Q-learning," in *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010, NIPS 2010*, Whistler, Canada, 2010.
- [19] B. Mavrin, H. Yao, and L. Kong, "Deep reinforcement learning with decorrelation," pp. 2094–2100, 2019, <http://arxiv.org/abs/1903.07765>.
- [20] S. Fujimoto, H. Van Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *35th International Conference on Machine Learning, ICML 2018, vol. 4*, pp. 2587–2601, Stockholmssmässan, Stockholm SWEDEN, 2018.
- [21] D. P. Kingma and J. L. Ba, "Adam: a method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015- Conference Track Proceedings*, The Hilton San Diego Resort & Spa, 2015.

- [22] Y. Wang and T. Ni, "Meta-SAC: auto-tune the entropy temperature of soft actor-critic via metagradient," 2020, <http://arxiv.org/abs/2007.01932>.
- [23] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," in *4th International Conference on Learning Representations, ICLR 2016- Conference Track Proceedings*, Caribe Hilton, San Juan, Puerto Rico, 2016.

Research Article

Deep Personalized Medical Recommendations Based on the Integration of Rating Features and Review Sentiment Analysis

Qihang Zhou, Lei Su , Liping Wu, and Di Jiang

Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China

Correspondence should be addressed to Lei Su; s28341@hotmail.com

Received 7 January 2021; Revised 30 April 2021; Accepted 17 May 2021; Published 28 May 2021

Academic Editor: Yin Zhang

Copyright © 2021 Qihang Zhou et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To comply with the rapid development of big data in mobile services, an increasing number of websites have begun to provide users with recommendation decisions in various areas, like shopping, tourism, food, and medical treatment. However, there are still some challenges in the field of medical recommendation systems, such as the lack of personalized medical recommendations and the problem of data sparseness, which seriously restricts the effectiveness of such recommendations. In this paper, we propose a personalized medical recommendation method based on a convolutional neural network that integrates revised ratings and review text, called revised rating and review based on a convolutional neural network (RR&R-CNN). First, the review text is divided into user and doctor datasets, and BERT vectorized representations are performed on them. Moreover, the original rating features are revised by adding the sentiment analysis values of the review text. Then, the vectorized review text and the revised rating features are spliced together and input into the convolutional neural network to extract the deep nonlinear feature vectors of both users and doctors. Finally, we use a factorization machine for feature interaction. We conduct comparison experiments based on a Yelp dataset in the “Health & Medical” category. The experimental results confirm the conclusion that RR&R-CNN has a better effect compared to a traditional method.

1. Introduction

With the advent of the age of information intelligence, mobile computing and recommendation systems are developing rapidly. However, in the face of massive amounts of information, the personalized demands of users are constantly increasing, and thus, the requirements for recommendation results and performance are also increasing. Traditional recommendations face difficulty in satisfying users' needs. Since deep learning can learn deep nonlinear feature representations, it is of great significance to study recommendation systems combined with deep learning [1, 2]. A recommendation system covers all aspects of people's lives. Correspondingly, recommendations for medical services are gradually developing. However, some problems cannot be ignored in the field of medical recommendation systems. (1) Currently, most medical software or platforms only provide an open interactive window for

users to communicate and seek medical treatment online. Moreover, due to the excessive number of experts and the complex needs of users, a platform cannot accurately recommend suitable doctors for users and lacks personalization recommendations [3]. (2) Like those in other fields, medical recommendation systems may also face the problem of sparse rating data [4] since the doctors evaluated by each user only account for a small part of the total number of doctors. The solution to this problem is to add review text [5], as it expresses a user's emotions and explains his/her rating. However, the accuracy of user reviews and ratings is interfered with by users' sentiments. Sometimes, a user provides a poor evaluation of a doctor but gives him/her a high rating. For example, we find a negative description in a 4-star review, which affects the objectivity and accuracy of the evaluation of this doctor. Therefore, it is necessary to combine the original rating with the sentiment analysis value of the review.

To solve the problem of the inaccuracy of rating and static coding of review text in a medical recommendation system, in this paper, the main contributions are shown as follows.

RR&R-CNN uses the revised rating and review text as input to the model. We use the revised rating combining the original rating and the review sentiment analysis to provide better and more objective recommendations. This is mainly considering that, sometimes, a user's ratings and evaluations of doctors do not match, and the user's review text expresses his/her emotional tendencies. Meanwhile, we use the pretrained model—BERT [6]—to obtain the vector representation of the review text. This model can resolve the problem of polysemy to help us understand the semantics and emotions in the review text and improve the accuracy of our rating prediction. In the input layer of the convolutional neural network model, a user's review text of doctors and the revised rating are combined as input features. We conduct experiments on a Yelp dataset, and the experimental results show that RR&R-CNN effectively improves the recommendation effect.

The remainder of this paper is structured as follows. Section 2 presents the related works on recommendation systems based on deep learning, the representation of review text and sentiment analysis, and their applications in recommendation systems. Section 3 describes the components of our network model in detail. In Section 4, we introduce the design of our experiments and the analysis of the experimental results. Finally, we conclude the paper in Section 5.

2. Related Work

2.1. Deep Learning-Based Recommender System. A recommendation system can help users deal with information overload and provide them with personalized suggestions, content, and services [7]. Traditional recommendation algorithms cannot learn the deep features of users and items by using shallow models. In recent years, deep learning has made rapid progress and development in many fields. It can not only perform deep nonlinear representations on massive user and item datasets but also automatically learn features from multisource heterogeneous data, especially for unstructured data such as text and pictures to extract feature information [8]. Thus, deep learning has become a crucial research hotspot in recommendation systems.

Recommendation systems based on deep learning usually take user and item-related data such as explicit and implicit feedback data as input features, and they generate item recommendations for users from the implicit representation of users and items obtained through the deep learning model [8]. A convolutional neural network is a typical model in deep learning. It is a special feedforward neural network with convolution and pooling operations that can effectively capture global and local features and significantly improve efficiency and accuracy [9]. Convolutional neural networks are widely used in recommendation systems, mainly for learning feature representations from multisource data such as images, text, audio, and video. For review text, convolutional neural networks have been widely used because of their pow-

erful feature extraction ability. Kim et al. [10] used convolutional neural networks to obtain deep hidden vectors from the text information of commodities and used multilayer convolution to obtain the correlation between words, which improved the accuracy of rating prediction in the recommendation system. Zheng et al. [11] proposed a parallel convolutional neural network that processes both user and product text information. A convolutional neural network extracts the deep features of users and products. Then, a factorization machine is introduced as the corresponding rating estimator.

2.2. Representation of Review Text and Its Application in Recommendation Systems. Review text usually contains information about a user's emotional expression and product features, which is of great reference value for other users in making decisions [12]. In medical recommendations, user reviews generally include detailed and personalized evaluations of doctors, such as feelings about the medical environment and doctors' abilities and attitudes [13]. In the rating prediction task, we need to extract the necessary information from the review text and convert it into a language that can be recognized by a computer. This task involves the representation method of review text, which converts the text into a vector representation. Widely used vector representation methods include word2vec [14], Glove [15], ELMo [16], and BERT [6].

The core idea of the word2vec model is to obtain a vectorized representation of a word through its context. Although the structure of this method is easy to understand, it can still learn high-quality word embeddings. Glove is based on word2vec considering the use of statistical information in the corpus. The above two methods of expressing word vectors have achieved certain results, but both of them belong to static coding; that is, the same word has the same meaning in different contexts. This may lead to a deviation in the understanding of semantics [17]. The ELMo model, which was proposed in early 2018, dynamically generates word vectors according to context. Its network structure adopts a "two-layer and two-way" long short-term memory (LSTM) network language model, which effectively alleviates the problem of polysemy. In October of the same year, the BERT model was born, which uses dual task pretraining and fine-tuning training strategies to perform pretraining on a large number of unlabeled texts. The BERT model has made substantial progress in 11 NLP tasks [6]. The difference between BERT and ELMo is that the representation of each word in the ELMo model is based on the current entire sentence, while the BERT model considers the full text of the context when generating word vectors. In [18], a pretrained BERT model is used to replace the previous word embedding layer, which can obtain a more accurate semantic understanding and further improve performance.

2.3. Sentiment Analysis and Its Application in Recommendation Systems. Emotional factors play a vital role in a user's decision-making process. The feedback provided by users in the face of different types of information reflects their emotional state at the time, and this change in emotional state can directly affect their thinking [19]. Users'

TABLE 1: An example of a user’s review text.

Text for 3 stars
I’ve come here very often and in the past it was great and I loved it but it seems like things have changed a lot. The staff in the front were very nice but the nurses and doctors were rude and impatient. The doctor did not listen to what I was saying and had to repeat myself often. He seemed angry at me the whole time and made me uncomfortable. If it wasn’t for good past experiences I would’ve given 2 stars.

emotions are expressed explicitly through product ratings and implicitly through product reviews. Therefore, a recommendation system can use emotional factors to analyze a user’s behavior and then judge his/her choice in terms of different items [20]. However, Zhang et al. [21] compared user ratings with their sentimental tendencies to review text and found that the former cannot completely replace the latter. For example, as shown in Table 1, the review text expresses the negative sentiment that doctors and nurses are both rude and impatient. A user’s rating of “3 points” is not a true evaluation, but an encouraging sense, which directly affects the accuracy of the doctor’s evaluation. Therefore, it is of great significance to study the combination of rating and review sentiment analysis.

At present, quite a few researchers have proposed related methods. In [22], Abbasi-Moud et al. used semantic clustering and sentiment analysis to extract user preference information from user review text. In [23], Zhao et al. calculated the sentiment deviation of the user’s review text and integrated it into the matrix decomposition to improve the accuracy of the score prediction. Lei et al. [24] proved that users’ social emotions, including users’ emotional similarity, interpersonal emotional influence, and project reputation, play a key role in improving rating prediction. In the location recommendation, Wang et al. [25] proposed a system that could recommend green space to users. They mainly calculated the contents of various tweets into the polarity values that could reflect users’ emotions by using the emotional-relevance dictionary and understood users’ attitudes towards them through the polarity values. Zhao et al. [26] proposed a method for mining points of interest that combines geographic attributes and emotional attributes. The emotional attributes are mainly based on emotional analysis of text information. In [27], Zheng et al. proposed a new preference prediction mechanism from the perspective of users’ subjective emotions, taking into account the influence of changes in user emotions over time on user behavior in sequence recommendation. In addition, Zhang et al. [28] proposed a method of prefiltering opinions using a comprehensive measurement of a user’s emotional orientation and original rating level. This method uses word2vec and LSTM to model the sentiment analysis of the review text to predict the sentiment tendency rating of each user’s reviews of an item, and then, it performs a weighted sum with the original rating to correct the deviation between the original rating and the user’s emotional tendency. The HFT model [29] uses matrix factorization to obtain the latent factors of the scoring value. Moreover, topic distribution was used to mine the latent

topics of review texts. Then, a mapping relationship between latent factors and latent topics was established to achieve model fusion.

3. RR&R-CNN: Medical Recommendation Model Based on a Convolutional Neural Network’s Integration of Rating and Review Sentiment Analysis

In this paper, we divide the review text into a user dataset and a doctor dataset. Then, we use a pretrained BERT model to obtain vectorized representation of the review text. Moreover, we combine the original rating data of the user and the doctor with the sentiment analysis value of the review text to revise the rating. Finally, the vectorized representation of the review text and the revised rating is combined as the input data of the convolutional neural network. Therefore, this method can discover the high-level features of users and doctors and provide personalized doctor recommendation services for users.

3.1. Model Structure. RR&R-CNN consists of the following three parts, and its specific architecture is shown in Figure 1.

- (1) Input layer: the review text is divided into a user review dataset $\{\text{User}T_1, \text{User}T_2 \cdots \text{User}T_d\}$ and a doctor review dataset $\{\text{Doctor}T_1, \text{Doctor}T_2 \cdots \text{Doctor}T_d\}$, and we use the pretrained BERT model to vectorize the review text; for the rating data, we use the sentiment analysis value of the review text combined with the original rating as the revised rating, and we splice it with the embedding value of the review text
- (2) Feature extraction layer: based on the feature extraction layer of the convolutional neural network, we use parallel CNN to extract the deep features of users and doctors. This layer includes the convolutional layer, pooling layer, and fully connected layer
- (3) Rating prediction layer: we interact the extracted features of the user and the doctor through the factorization machine to obtain the predicted value, and we use the loss function to calculate the error between the predicted value and the true value

3.2. Preprocessing of the Input Layer. We perform sentiment analysis and processing on the review text to obtain the sentiment polarity value. We use the TextBlob sentiment analysis tool, which is a tool used to perform natural language processing tasks. The range of its emotional polarity is $[-1,1]$, where -1 denotes negative, and 1 denotes positive. We use a formula to combine the emotional polarity value obtained from the review text with the original rating [1–5]. The specific combined Equation (1) is as follows [30]:

$$RS_{ij} = \rho R_{ij} + (1 - \rho) S_{ij}, \quad (1)$$

where R_{ij} represents the original rating of doctor j by user i ,

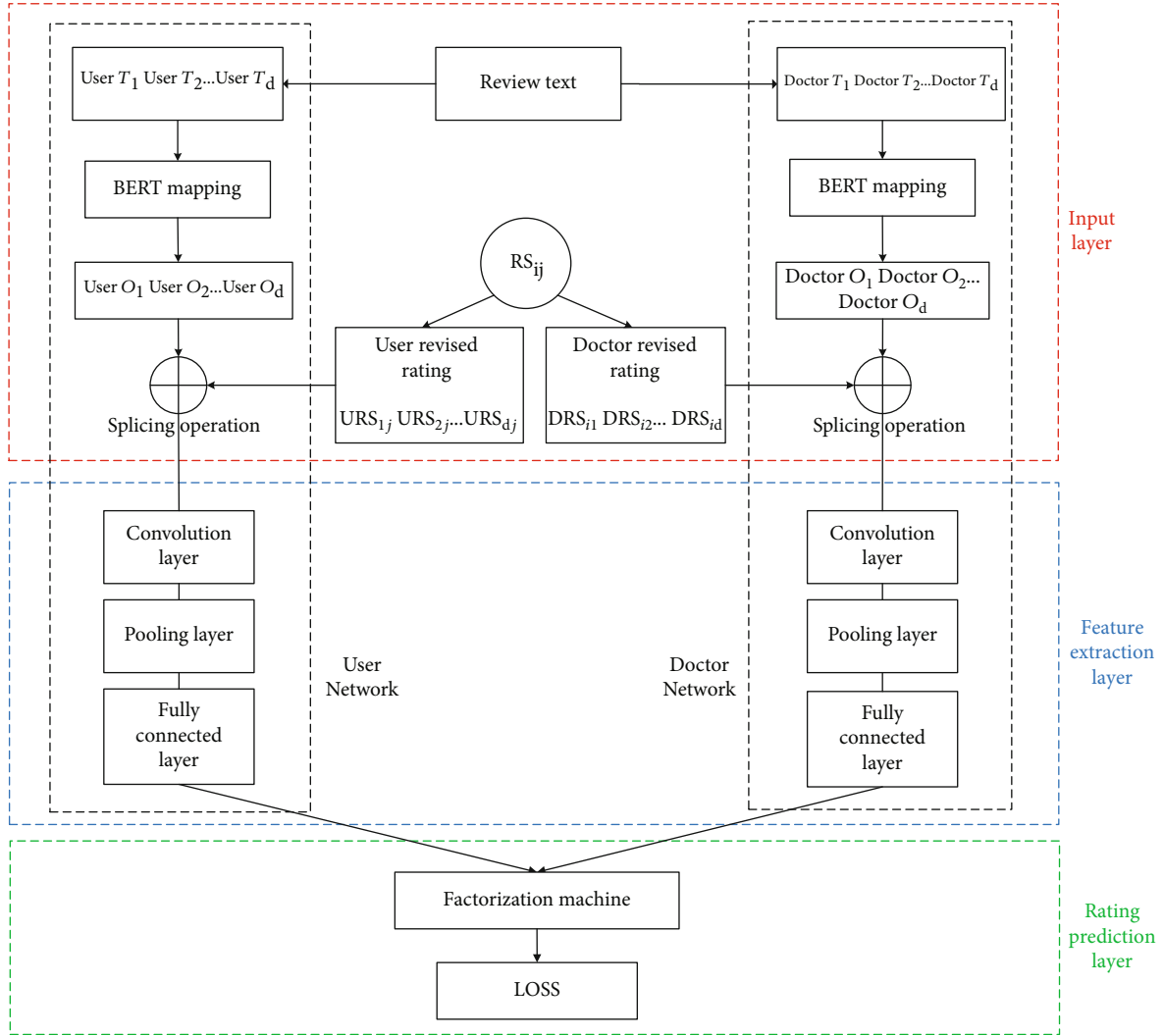


FIGURE 1: Model structure.

and the value is [1-5]. S_{ij} represents the value of the sentiment polarity of the review text calculated by using TextBlob. RS_{ij} represents the final rating after combining the sentiment analysis value of the review text with the original rating. ρ is a parameter used to adjust the weight between the original rating and the sentiment polarity value.

We load the pretrained BERT model into RR&R-CNN as a part of the model to update the vector representation of the review text. For user reviews, we convert the user review text $\{UserT_1, UserT_2 \dots UserT_d\}$ (d as the maximum number of reviews) into a vector $\{UserO_1, UserO_2 \dots UserO_d\}$ after being mapped by BERT. The number of embedding dimensions is 768. For doctor reviews, we use the same method. An important reason for using BERT is that it can solve the polysemy problem that cannot be solved by word2vec.

Finally, we use the direct splicing method, that is, splicing the embedded vector corresponding to a sentence with the corresponding revised rating. Compared with the original input dimension, one-dimensional data are added, and the width of the convolution kernel is increased by a factor of

1. We splice the vectorized review texts with the revised ratings as the input of the convolutional neural network.

3.3. Feature Extraction Layer. The feature extraction layer mainly uses a convolutional neural network to extract deep features, including a convolutional layer, pooling layer, and fully connected layer.

3.3.1. Convolutional Layer. After splicing the sentence vector obtained by BERT with the revised rating feature, the convolution kernel is used to convolve it. The convolutional layer is composed of convolution kernels of different sizes. Because the weights of the convolution kernels are shared, a convolution kernel can extract only a certain type of feature, so we need to use different sizes of convolution kernels, which represent different feature extractors. After calculating a convolution kernel, a column vector is obtained, which represents the features extracted by the convolution kernel from the sentence. Each convolution kernel performs a convolution operation to obtain a feature map. The use of the convolution

operation and the activation function to obtain the corresponding feature map is expressed as Equation (2):

$$c_i = f(w * UO_{i:i+t-1} + b), \quad (2)$$

where c_i represents the operation result of the convolution kernel. w is the weight of the convolution kernel. b is the bias term of the convolution kernel. $UO_{i:i+t-1}$ represents the vector matrix. $*$ is the convolution operation. $f(x)$ is the activation function. In this paper, we use the rule activation function. The representation of all feature maps C (3) obtained after the convolution operation is as follows:

$$C = \{C_1, C_2, \dots, C_{d-t+1}\}. \quad (3)$$

3.3.2. Pooling Layer. Since we use convolution kernels with different heights in the convolution layer, the vector dimensions obtained after convolution will be different. The pooling layer follows the convolution layer, which aims to reduce the number of features, and it decreases the amount of calculation required by reducing dimensions. In this paper, we use the max-pooling layer. The principle of the max-pooling layer is to select the largest value in the feature map. This maximum value represents the most important feature that we capture and the final feature that we want to select. The new feature map discards other unimportant features. Equation (4) is as follows:

$$C = \max(C). \quad (4)$$

After the pooling operation, we obtain a fixed-size feature representation.

3.3.3. Fully Connected Layer. We input the results of the pooling layer into the fully connected layer, multiply it by the weight matrix of the fully connected layer, add a bias, and adopt a random dropout strategy, that is, randomly deleting the neurons in the neural network. In each iteration, we only use part of the neuron training model to obtain the values of w and b and lose approximately half of the neurons each time, which is equivalent to training on different neural networks. Thus, the dependence of neurons is reduced to solve the problem of overfitting.

3.4. Prediction Layer. Although the above outputs can be regarded as the features of users and items, these two outputs may not be directly compared in different feature spaces. Therefore, a prediction layer is set at the end of the model to couple U_u and N_i , and the features of user u and item i are spliced into a single vector, $z = (u, i)$. We introduce the factorization machine [31] to perform feature interaction, and the user's prediction rating for doctors is obtained after multiple trainings. Given N training samples, Equation (5) is used for training:

$$P = w_0 + \sum_{i=1}^n w_i z_i + \sum_{i=1}^n \sum_{j=i+1}^n z_i z_j \sum_{f=1}^k v_{i,f} v_{j,f}, \quad (5)$$

We use the loss function of the tf.nn.L2_loss model to train

the target. This function uses the L2 norm to calculate the error value of the tensor. Equation (6) is as follows:

$$\text{Loss}(\text{prediction}, \text{input}_y) = \frac{1}{n} \sum |\text{prediction} - \text{input}_y|^2, \quad (6)$$

where input_y represents the true value, and prediction represents the predicted value.

4. Experimental Results and Analysis

4.1. Experimental Data Set and Evaluation Metrics. This experiment uses a Yelp dataset (<https://www.yelp.com/dataset>), which has a total of 66,859 million data points. We randomly select 1,000,000 pieces of data, and then, we find, screen, and analyze the "Health & Medical" data according to the category from these 1,000,000 pieces of data. Finally, the dataset used in this experiment is selected from the medical-related classification dataset, which contains 69,813 reviews made by 57,953 users about 4,658 doctors. The density of the experimental dataset is 0.026%. In the experiment, 80% of the dataset is randomly selected as the training set, 10% is regarded as the validation set, and the other 10% is treated as the test set. The experiment was repeated three times to obtain the average value.

The root mean square error (RMSE) is used as the standard to evaluate the performance of the algorithm, as it evaluates model accuracy by calculating the difference between the predicted rating and true rating, and it is one of the most commonly used evaluation indicators in recommendation systems. Generally, the smaller the value of the RMSE is, the better the recommended effect. Equation (7) is as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{n=1}^N (r_n - \hat{r}_n)^2}. \quad (7)$$

In the above formula, \hat{r}_n represents the n th predicted rating, r_n represents the n th true rating, and N represents the number of samples.

4.2. Comparison Method. We select two traditional methods—the latent factor model (LFM) [32] and latent Dirichlet allocation (LDA) [21]—and three deep learning methods—deep cooperative neural networks (DeepCoNN) [11], convolutional matrix factorization (ConvMF) [10], and variants of DeepCoNN (DeepCoNN+rating). First, we select the LFM model that only contains rating features but not the review text information as comparative experiments. The LFM uses matrix factorization to predict the rating of unknown commodities. Second, we select the LDA model that only contains review text information as comparative experiments. LDA is used to learn the document-topic distribution matrix from the review text as the potential feature of the item, and the optimal solution is found through gradient descent for recommendation. Third, we select the DeepCoNN model, a deep learning model, for comparative experiments. DeepCoNN convolutes the user review text and the commodity review text to predict the rating. Fourth, we select a deep learning model, ConvMF, involving ratings and reviews. This

model combines CNN and probabilistic matrix factorization (PMF) to predict rating. Fifth, we add a revised rating feature to the DeepCoNN to obtain the “DeepCoNN + rating” algorithm, which combines the original rating with the sentiment analysis rating of the review, and we use this algorithm to perform comparative experiments. Finally, we improve the fifth method by using the pre-trained BERT model instead of word2vec static encoding for the review text, which can effectively obtain semantic information, and it also revises the original rating by performing sentiment analysis on the review text. This method can effectively improve recommendation performance. The difference between RR&R-CNN and DeepCoNN is that the former uses BERT to vectorize the review text, while the latter uses word2vec; moreover, the former adds the revised rating features that are spliced with the embedded values of the review text to the input layer and inputs them into the convolutional network, whereas the latter only uses review data in the input layer.

4.3. Experimental Environment and Parameter Settings. The experimental environment configuration of this paper is as follows. The system version is Ubuntu 18.04, the GPU model is RTX2080Ti/11 GB, and the CUDA version is 10.2. In addition, the deep learning framework is TensorFlow 1.14. The specific experimental parameter configuration is as follows.

4.3.1. Embedded Representation. DeepCoNN uses word2vec pretrained vectors—Google News [33]—to embed the review text. The word vector is approximately 100 billion words trained from Google News, containing 3 million words and phrases, and the dimension of the vector is 300. The continuous word bag structure model is used for training, and the vectors are not updated during the training. If there are words in the sentence that are not in the pretrained dictionary, then, they will be used instead of random numbers. ConvMF uses the word embedding word2vec. RR&R-CNN uses the pretrained BERT model—“BERT-Base, cased_L-12_H-768_A-12.” This model uses a 12-layer transformer, the parameter of multihead attention is 12, the total parameter size of the model is 110 MB, the maximum length of the selected review text is 256, and the embedding dimension of the word vector is 768. The optimizer uses Adam, and the initial learning rate is $\mathbb{N} - 5$.

4.3.2. Parameter Setting of Networks. The RR&R-CNN convolution kernel size is $t * v$, where t represents the number of words contained in the vertical direction of the convolution kernel, the values are 3, 4, and 5, and each size has 100 convolution kernels. v is the dimension of the embedding word vector. To prevent the model from overfitting, we add a dropout operation, in which the dropout rate is 0.2, to the network structure of the model. The number of latent factors for users and doctors in the factorization machine is 64 from {16, 32, 64, 128}. The L2 regularization parameter in the loss function is 0.6. The values of the parameters in DeepCoNN and ConvMF are consistent with those in the original paper.

4.3.3. Traditional Model Parameters. In the comparative experiment of the LDA model, which only uses the review text data, theme dimension k is set to 10, superparameter α is 0.2, and β is 0.1. In the comparative experiment of the LFM that only uses rating data, the number of hidden factors is set to 20, the regularization coefficient is 0.01, and the learning rate is 0.1.

4.4. Experimental Results

4.4.1. Comparison of Experimental Results. The experimental parameters of RR&R-CNN are as follows: ρ is the weight between the revised original rating and the sentiment polarity value. The value range of ρ is {0, 0.1, ...0.9, 1}, and 0.8 is the selected value that obtains the lowest RMSE. The number of convolution kernels is set to 100 from {100, 200, 300, 400}. The number of iterations is 6. The value of hyperparameter K in the factorization machine is set to 32 from {16, 32, 64, 128}. Table 2 shows a comparison of the experimental results.

The experimental data in Table 2 are repeated three times, and the average value is taken. At the same time, the best results are expressed in *italic*. The experimental results show that in the traditional method, the effect of the LDA model, which only considers the review text, is worse than that of the LFM, which only uses the rating. This shows that only considering the review text does not guarantee that the learned features will be helpful for rating prediction, which means it cannot be completely independent of rating features. Moreover, the performance of deep learning models is better than that of traditional methods, which demonstrates the powerful advantages of deep learning. DeepCoNN model considers the review text of the user and doctor. The ConvMF model uses only the review text of the doctor but adds the rating information of the user, and its effect is better than that of DeepCoNN, which indicates that combination of rating features and comment text is indeed better than using the comment text alone. RR&R-CNN based on the deep learning model, combined with the review text and the revised rating features, reduces prediction error by 4.2% compared with DeepCoNN, which proves that the performance of our recommendation model is better.

To further prove the effectiveness of each module in RR&R-CNN, we show the ablation study—DeepCoNN+rating and DeepCoNN-BERT. In Table 3, DeepCoNN is the baseline, DeepCoNN+rating adds the revised rating feature on the basis of the baseline, and DeepCoNN-BERT replaces word2vec with BERT based on the baseline. The experimental results show that the effect of adding the revised rating feature and using BERT to embed the review text simultaneously is better than that of using either method alone. It demonstrates the combination of the two factors is effectiveness.

4.4.2. The Value of ρ . As shown in Figure 2, we set the value of ρ from 0 to 1, with a step size of 0.1. When ρ is 0, it means that only the emotional polarity value of a user’s review text is considered; when $\rho = 1$, it means that only

TABLE 2: Comparison of the experiment results from the Yelp dataset.

Algorithm	RMSE
LDA	1.784
LFM	1.689
DeepCoNN	1.653
ConvMF	1.647
DeepCoNN+rating	1.618
RR&R-CNN	1.583

TABLE 3: The ablation study.

Algorithm	RMSE
DeepCoNN	1.653
DeepCoNN-BERT	1.643
DeepCoNN+rating	1.618
RR&R-CNN	1.583

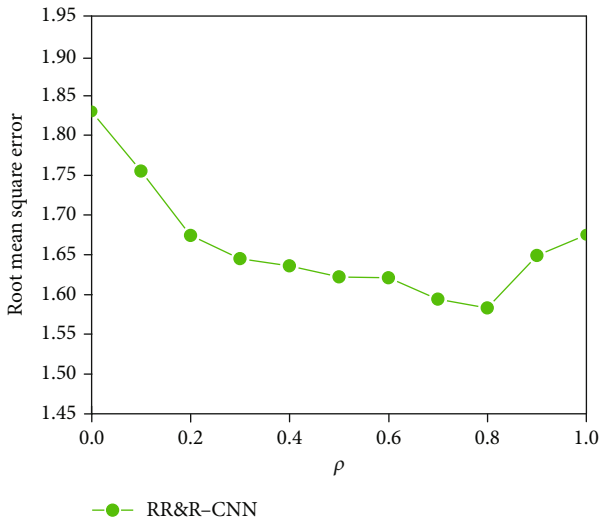


FIGURE 2: RMSE variation with different ρ values.

a user’s original rating is considered. It can be seen from the data displayed in the figure that when ρ is 0.8, the RMSE value is the smallest. When ρ is greater than 0, the corresponding RMSE value decreases to various degrees. This confirms our hypothesis that the use of a combination of the original rating and the sentiment analysis value of the review text, instead of the original rating data, can effectively alleviate the problem in which users sometimes have a low evaluation of doctors but give them high ratings.

4.4.3. *The Number of Convolution Kernels.* As shown in Figure 3, when the value of ρ is 0.8, the number of selected convolution kernels ranges from 10 to 400. When the number of convolution kernels is 100, the RMSE value is the lowest, and when the number of convolution kernels is greater than 100, the model shows an upward trend; that is, the per-

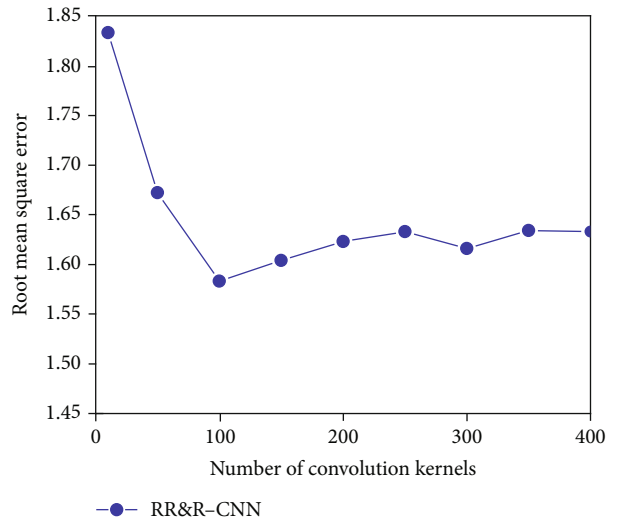


FIGURE 3: RMSE variation with different numbers of convolution kernels.

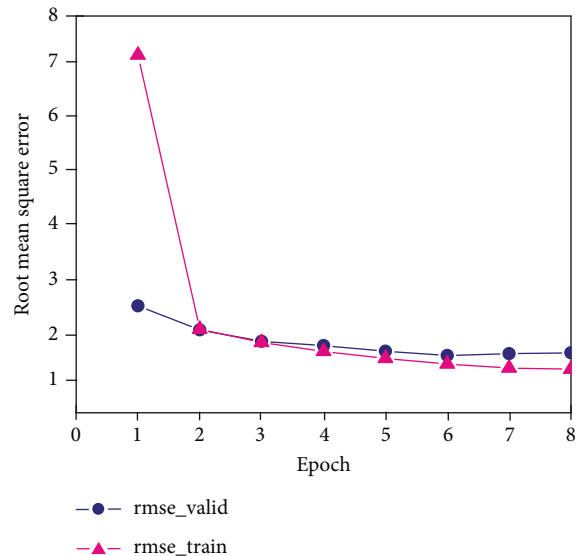


FIGURE 4: RMSE variation with different numbers of iterations.

formance becomes worse, so the number of convolution kernels is set to 100.

4.4.4. *The Number of Iterations.* As shown in Figure 4, we perform various experiments with different iterations. When the epoch is 6, the value of RMSE is the lowest, and both $rmse_{train}$ and $rmse_{valid}$ continuously decrease. When the number of epochs is greater than 6, $rmse_{train}$ decreases continuously, while $rmse_{valid}$ increases instead of decrease, indicating that the model only fits $rmse_{train}$ very well, so the epoch value is 6.

4.4.5. *The Value of FM_k .* As shown in Figure 5, the experiment is carried out for different K values {16, 32, 64, 128}. When the fixed epoch is selected as 6, the k value is 32, and the RMSE value is the lowest.

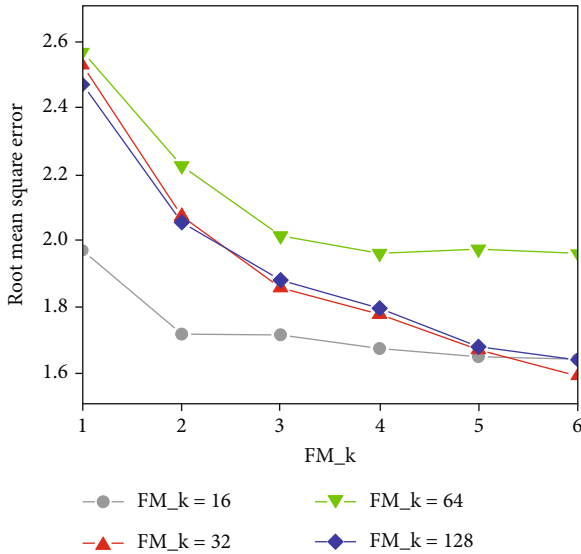


FIGURE 5: RMSE variation with different FM_k values.

5. Conclusions

To solve the problem of how to better recommend doctors according to the personalized demands of users, we propose RR&R-CNN. (1) In contrast to the traditional distributed text representation, RR&R-CNN uses the pretrained BERT model to obtain the vectorized representation of the review text, which solves the problem of polysemy. (2) Regarding the rating features, considering the reasons that, sometimes, a user's evaluation does not match his/her rating, we use the revised user and doctor ratings as the final representation. The revised rating and the representation of the review text are spliced together as the input of the convolutional neural network for deep feature extraction. (3) As the model is divided into a user network and a doctor network, the factorization machine is used to perform the feature interaction between the two networks, and the loss function is used to reduce the calculation error in rating prediction.

The experimental results show that the recommendation effect is improved after adding the revised rating features and that the BERT vectorization of the review text is better than word2vec vectorization. However, a disadvantage of the BERT model is that the time and memory consumed by it are very large. At present, there are some methods for compressing and reducing the complexity of the BERT model. In future work, we will verify this. If it indeed reduces the time and memory consumed by the experiment and improves the recommendation performance, then, we will improve it within the existing method. In addition, BERT, which involves knowledge in the medical field, such as ClinicalBERT [34] and BioBERT [35], will also serve as our follow-up research work.

Data Availability

This experiment uses the Yelp dataset (<https://www.yelp.com/dataset>).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The work is supported by the National Science Foundation of China under Grant no.61365010.

References

- [1] R. Mu, "A survey of recommender systems based on deep learning," *IEEE Access*, vol. 6, pp. 69009–69022, 2018.
- [2] Z. Batmaz, A. Yurekli, A. Bilge, and C. Kaleli, "A review on deep learning for recommender systems: challenges and remedies," *Artificial Intelligence Review*, vol. 52, no. 1, pp. 1–37, 2019.
- [3] M. S. Hossain, "Cloud-supported cyber-physical localization framework for patients monitoring," *IEEE Systems Journal*, vol. 11, no. 1, pp. 118–127, 2015.
- [4] H. Wang, N. Wang, D. Yeung et al., "Collaborative deep learning for recommender systems," in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1235–1244, August 2015.
- [5] G. Ling, M. R. Lyu, and I. King, "Ratings meet reviews, a combined approach to recommend," in *2014 in Proceedings of the 8th ACM Conference on Recommender systems - RecSys '14*, pp. 105–112, 2014.
- [6] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "Bert: pre-training of deep bidirectional transformers for language understanding," 2018, <http://arxiv.org/abs/1810.04805>.
- [7] G. Adomavicius and A. Tuzhilin, "A toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, 2005.
- [8] L. W. Huang, B. T. Jiang, S. Y. Lv, Y. B. Liu, and D. Y. Li, "Survey on deep learning based recommender systems," *Chinese Journal of Computers*, vol. 41, no. 7, pp. 1619–1647, 2018.
- [9] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system," *ACM Computing Surveys*, vol. 52, no. 1, pp. 1–38, 2019.
- [10] D. Kim, C. Park, J. Oh, S. Lee, and H. Yu, "Convolutional matrix factorization for document context-aware recommendation," in *Proceedings of the 10th ACM conference on recommender systems*, pp. 233–240, September 2016.
- [11] L. Zheng, V. Noroozi, and P. S. Yu, "Joint deep modeling of users and items using reviews for recommendation," in *Proceedings of the tenth ACM international conference on web search and data mining*, pp. 425–434, 2017.
- [12] Y. Wang, M. Wang, and W. Xu, "A sentiment-enhanced hybrid recommender system for movie recommendation: a big data analytics framework," *Wireless Communications and Mobile Computing*, vol. 2018, Article ID 8263704, 9 pages, 2018.
- [13] Y. Qian, Y. Zhang, X. Ma, H. Yu, and L. Peng, "EARS: emotion-aware recommender system based on hybrid information fusion," *Information Fusion*, vol. 46, pp. 141–146, 2018.
- [14] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *International Conference on Learning Representations*, 2013.

- [15] J. Pennington, R. Socher, and C. D. Manning, "Glove: global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, Doha, Qatar, October 2014.
- [16] M. E. Peters, M. Neumann, M. Iyyer et al., "Deep contextualized word representations," in *North American Chapter of the Association for Computational Linguistics*, pp. 2227–2237, 2018.
- [17] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, and X. Huang, "Pre-trained models for natural language processing: a survey," *Science China Technological Sciences*, vol. 63, pp. 1872–1897, 2020.
- [18] X.-J. Feng and Y.-Z. Zeng, "Joint deep modeling of rating matrix and reviews for recommendation," *Chinese Journal of Computers*, vol. 43, no. 5, pp. 884–900, 2020.
- [19] M. Tkalcic, A. Kosir, and J. Tasic, *Affective Recommender Systems: The Role of Emotions in Recommender Systems*, CEUR-WS.org, 2011.
- [20] G. Gonzalez, J. L. De La Rosa, M. Montaner, and S. Delfin, "Embedding emotional context in recommender systems," in *2007 IEEE 23rd International Conference on Data Engineering Workshop*, pp. 845–852, Istanbul, Turkey, April 2007.
- [21] Y. Zhang, H. Zhang, M. Zhang, Y. Liu, and S. Ma, "Do users rate or review? Boost phrase-level sentiment labeling with review-level sentiment classification," in *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pp. 1027–1030, July 2014.
- [22] Z. Abbasi-Moud, H. Vahdat-Nejad, and J. Sadri, "Tourism recommendation system based on semantic clustering and sentiment analysis," *Expert Systems with Applications*, vol. 167, article 114324, 2021.
- [23] G. Zhao, X. Lei, X. Qian, and T. Mei, "Exploring users' internal influence from reviews for social recommendation," *IEEE Transactions on Multimedia*, vol. 21, no. 3, pp. 771–781, 2018.
- [24] X. Lei, X. Qian, and G. Zhao, "Rating prediction based on social sentiment from textual reviews," *IEEE Transactions on Multimedia*, vol. 18, no. 9, pp. 1910–1921, 2016.
- [25] J. Wang, Y. Feng, E. Naghizade, L. Rashidi, K. H. Lim, and K. Lee, "Happiness is a choice: sentiment and activity-aware location recommendation," in *Companion of the The Web Conference 2018 on The Web Conference 2018 - WWW '18*, pp. 1401–1405, 2018.
- [26] G. Zhao, P. Lou, X. Qian, and X. Hou, "Personalized location recommendation by fusing sentimental and spatial context," *Knowledge-Based Systems*, vol. 196, p. 105849, 2020.
- [27] L. Zheng, N. Guo, W. Chen, J. Yu, and D. Jiang, "Sentiment-guided sequential recommendation," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1957–1960, July 2020.
- [28] Y. Zhang, X. Zhu, C. Xu, and S. Dong, "Hybrid recommendation approach based on deep sentiment analysis of user reviews and multi-view collaborative fusion," *Chinese Journal of Computers*, vol. 42, no. 6, pp. 1316–1333, 2019.
- [29] J. McAuley and J. Leskovec, "Hidden factors and hidden topics: understanding rating dimensions with review text," in *Proceedings of the 7th ACM conference on Recommender systems*, pp. 165–172, October 2013.
- [30] Y. Zhang, M. Chen, D. Huang, D. Wu, and Y. Li, "iDoctor: personalized and professionalized medical recommendations based on hybrid matrix factorization," *Future Generation Computer Systems*, vol. 66, pp. 30–35, 2017.
- [31] H. Guo, R. TANG, Y. Ye, Z. Li, and X. He, "Deep FM: a factorization-machine based neural network for CTR prediction," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, August 2017.
- [32] Y. Koren, R. M. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.
- [33] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in Neural Information Processing Systems*, vol. 26, pp. 3111–3119, 2013.
- [34] K. Huang, J. Altosaar, and R. Ranganath, "Clinical BERT: modeling clinical notes and predicting hospital readmission," in *CHIL'20: ACM Conference on Health, Inference, and Learning: Workshop Track*, 2020.
- [35] J. Lee, W. Yoon, S. Kim et al., "BioBERT: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 2020.