Characterization of Traffic Flows in a Low-Resource Community Wireless Network

1st Emmanuel Ackerson*
Department of Computer Science
University of Cape Town
Cape Town, South Africa
ackemm001@myuct.ac.za
*Corresponding author

2nd Josiah Chavula
Department of Computer Science
University of Cape Town
Cape Town, South Africa
josiah.chavula@uct.ac.za

***Abstract*—Community Wireless Networks (CWNs) play a pivotal role in bridging the digital divide between urban and underserved areas, forming a crucial component of local socio-economic infrastructure. However, the unique architecture of these networks hinders their ability to adjust to growing technological advancements. While traffic analysis is essential for uncovering underlying network dynamics and facilitating performance optimization, most current research tends to focus on conventional networks, leaving CWNs understudied. This study presents a statistical analysis of traffic dynamics in a low-resource CWN, leveraging flow-level metrics to characterize the network behaviour. We classified traffic into eight distinct application categories, including Web Services, Video Streaming, and P2P File Transfer, and analyzed flow-level metrics such as unique application counts, flow volume, packet count, and byte size across multiple time scales. The findings reveal significant heterogeneity, with both application type and time shaping the network usage. Temporal analysis shows distinct weekday-weekend patterns, with business-oriented applications peaking during work hours. This investigation underscores the need for time-aware traffic management and QoS in resource-constrained networks, while providing a foundation for targeted policy design, intelligent monitoring, security and scalable infrastructure planning**

***Index Terms*—Low-resource Networks, Community Wireless Networks, Traffic characterization, Network Application categories, iNethi Community Wireless Network, Empirical Analysis of dataset, Flow-Level Statistical Analysis**

I. Introduction

The modern internet is sustained by an integrated architecture comprising both high-resource and low-resource networks. Their complementary roles ensure global connectivity, scalability, and the efficient delivery of services, ranging from basic internet access in rural areas to advanced digital platforms in high-demand urban centers [1][2][3]. High-resource, often conventional networks, operate in well-resourced environments, thus employ robust infrastructure, including high-capacity routers, multi-layer switches, and redundant backbone links to support high-throughput and mission-critical services [4]. National Research and Education Networks (NRENs), 5G and telecommunications backbones are but a few examples [5]. Conversely, low-resource functions under significant limitations, whether in computational capability, memory, bandwidth, energy efficiency, or hardware availability [6][7][8]. Such networks are essential in scenarios where high-resource infrastructure is either economically unfeasible or physically impractical. A typical example is Community Wireless Networks (CWNs), which play a crucial role in enabling remote and localized communication. Community Wireless Networks are grassroots initiatives where communities collaboratively construct and maintain wireless infrastructure using cost-effective IEEE 802.11 a/b/n technologies [9]. CWNs frequently operate in the unlicensed 2.4 GHz and 5 GHz ISM bands and are increasingly augmented with fiber-optic links to improve backbone connectivity [10]. These networks are characterized by their technical architecture and governance models and often operate as micro-Internet Service Providers (micro-ISPs), managed by members who co-develop and sustain services for local consumption. CWNs effectively bridge urban-rural digital divides, creating social and economic opportunities [11]. Notable examples includ Ocean View community network (iNethi) [12] and Zenzeleni.net [13](Dumane, n.d.) in South Africa, Linknet in Zambia [14], Freikfunk in Germany, Athens Wireless Metropolitan Network (AWMN) in Greece, Ninux.org in Italy and Guifi.net in Spain [15][9]. Unlike conventional Wireless ISPs (WISPs), CWNs generally function on a nonprofit basis, with an emphasis on open infrastructure, cooperative maintenance, and community-driven innovation, thereby fostering digital digital inclusion, experimentation and research [11][16][17]. However, the specific architecture of CWNs make it difficult to keep pace with the growing technological advancements. For instance, their dynamic and unpredictable topology, coupled with service disruptions complicate the implementation of conventional quality of service (QoS) mechanisms such as DiffServ and IntServ, as well as security measures. To help address these challenges, traffic characterization is employed to understand the network traffic dynamics [18]. Traffic characterization entails collection and analysis of traffic flow metrics across network nodes, applications, and devices to identify usage trends, performance criteria, and potential challenges [19]. Such insights aid network managers in making well-informed decisions to improve QoS, security, and ensure optimal resource allocation [20]. Considerable research has been conducted on network traffic analysis within conventional networks, which are widely used and are characterized by their sufficiently endowed computing

resources [21]. Conversely, comparable volume of studies is unseen for resource-constrained networks, in particular CWNs, a concern echoed by [10][22]. Consequently, this has led to difficulties in formulating targeted solutions that can support their accelerated growth. This problem, if left unattended, could undermine efforts aimed at fostering digital inclusion in underserved communities. While the exact reason for this discrepancy remains uncertain, a possible explanation could be attributed to their less structured nature, heterogeneity, low economic viability, and unique operational characteristics as opposed to widely used conventional networks. This study presents a case analysis of traffic behavior pertaining to low-resource networks, focusing on CWNs. We aim to identify distinct characteristics of these networks, and to enable the establishment of targeted network optimization policies. Using production dataset, application flows are classified into eight distinct categories, and analyzed based on the number of unique applications, flow volume, packet count, and byte size. These metrics are evaluated across bi-weekly, weekly, daily, and hourly timescales to uncover usage patterns and temporal trends in network behaviour.

The key contributions of this work are as follows: (1) The use of eight application categories to analyze and understand traffic dynamics in low-resource networks, (2) Identification and characterization of context-aware traffic patterns specific to low-resource Community Wireless Networks (CWNs), (3) The establishment of a foundation for formulating targeted optimization policies tailored to CWNs, and (4) Design of a scalable, deterministic, and transparent traffic classification mechanism leveraging rule-based strategies derived from nDPI analysis and destination port number heuristics.

A. iNethi Community Wireless Networks

iNethi Community Wireless Network is located in Ocean View, a suburb of Cape Town, South Africa. The iNethi CWN is a community-driven network of 22 access points serving approximately 20,000 residents across 1.75 square kilometers, with local participation in both service delivery and infrastructure management at the time of data collection. The network employs a mesh architecture utilizing TV White Space (TVWS) and WiFi radios to establish an auto-configurable network. iNethi CWN facilitates decentralized content distribution and supports local content creation, including applications and services hosted within the community. It operates all cloudlet services as secure Docker or LXD containers, utilizing a unified single sign-on API for seamless access to all services, as shown in Figure 1 [23]. Data is encrypted and accessible solely by the owner, designated users, or through a group master key in the event of a security breach. The system synchronizes data on local iNethi servers with the global iNethi network, as well as with other iNethi communities, accommodating users who transition between different iNethi networks. Furthermore, the iNethi CWN integrates with community wireless infrastructures, such as WiFi, TVWS, and OpenCellular, enabling users to engage with iNethi services via web browsers, mobile applications, as well as allowing IoT devices to connect through machine-to-machine (M2M) protocols.

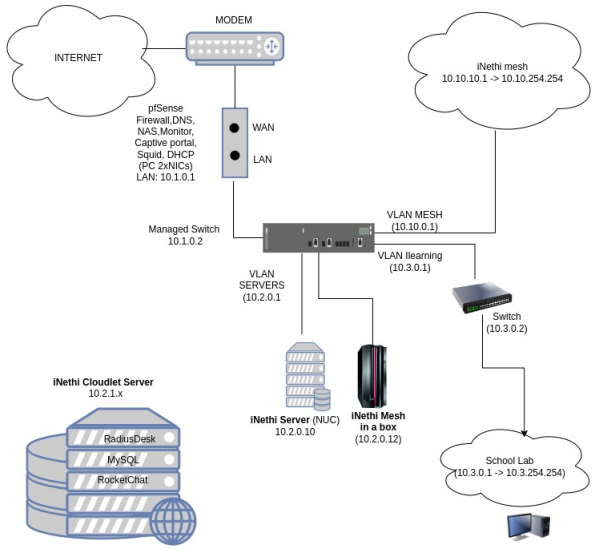

Fig. 1: Overview of iNethi CWN Architecture

II. Related Work

This section reviews prior research on traffic characterization across conventional and community wireless networks. Velan et al. [19] proposed a cross-layer profiling framework spanning IP, transport, and application layers. Their study across five university networks revealed consistent diurnal and weekly traffic patterns, low correlation between metrics, and effective differentiation between network types, highlighting the analytical value of diverse features. Frangoudis et al. [10] analyzed traffic from Linknet, a rural Zambian network using satellite and mesh links. Social media dominated usage, and OS updates consumed significant bandwidth. Performance issues were linked to the gateway and mesh infrastructure, with mitigation strategies including smart caching, off-peak scheduling, and data ferries. Maccari et al. [24] studied ninux.org, an Italian community network, and found a ”robust-but-fragile” dynamic, resilience coexisting with structural vulnerabilities. They emphasized the importance of regular structural assessments to detect anomalies. In a related study [25], Maccari and colleagues examined three large Wireless Community Networks (WCNs), assessing topology, node centrality, and the role of Multipoint Relays (MPRs) in OLSR-based routing. Their findings showed that WCNs are generally stable with efficient routing, and MPRs reduce signaling overhead without degrading performance. However, WCNs remain vulnerable to attacks due to architectural reliance on key nodes. Fomenkov et al. [26] analyzed Internet traffic at scale using packet header samples. They found that short sampling intervals obscured long-term patterns, and that average packet size scaled with traffic volume. Protocol usage varied across networks, highlighting heterogeneity in Internet traffic behavior. Jurkiewicz et. al [27] proposed a flow-based modeling

approach using extensive flow records from a university's Internet interface. They introduced accurate statistical models of flow size and packet distributions, recommending these as benchmarks for evaluating flow-based algorithms. Lastly, Malyeyeva et al. [28] performed a cross-country traffic analysis using nonparametric statistical tests and time series modeling, in his work. The study identified clear daily and annual periodic trends, with autoregressive models showing strong forecasting potential for long-term traffic behavior. These studies collectively underscore the importance of multi-layered analysis, temporal granularity, and architectural awareness in understanding and optimizing network traffic.

III. Methodology

In this study, we perform an empirical analysis of real-world CWN data, aiming to identify network-specific characteristics, and to establish a foundation for context-aware optimization policies. Our employed methodology begins with the collection of a traffic dataset from the iNethi Community Wireless Network. Following that, flows are extracted from the raw trace files using a packet-to-flow conversion technique, where application and service types within the traffic are then identified and labeled. The raw data undergoes preprocessing to ensure it is cleaned and structured, resulting in a human- and machine-readable format in comma-separated values (.csv). Identified applications are classified into eight (8) distinct categories, from which both application-level and flow-level statistics are derived. A detailed analysis is conducted to uncover key characteristics that distinguish traffic patterns, including underlying causes and their potential impact on network performance. The section concludes with a summary of the key insights obtained and recommendations

A. Network Traffic Data Collection

The iNethi CWN has been established to fulfill two primary objectives: (a) providing access to local content and services, and (b) facilitating Internet connectivity through a RADIUS authentication system [23][12]. WiFi coverage is achieved via twenty-two (22) mesh nodes distributed throughout the community, delivering a downlink speed of 10 Mbps and an uplink speed of 5 Mbps, utilizing 5GHz backhaul radios connected to fiber infrastructure located 10 kilometers away. We captured the network traffic dataset from the operational network (iNethi community) comprising two (2) VLANS, using the tcpdump toolset at the network gateway. The VLAN-1 consists of residential subscribers (iNethi Focus) on a monthly contract consumption basis, while VLAN-2 includes other consumers (INethi OVap) accessing services on an on-demand basis. Tcpdump is an open-source Linux tool for real-time packet capture and analysis, offering detailed insights into network traffic [29]. It captures transport layer data (e.g., TCP/UDP headers), application layer payloads, and key header information, including MAC addresses, IP addresses, port numbers, protocol types, and flags, spanning the data link, network, and transport layers. We configured tcpdump to automatically capture traffic at hourly intervals, thus recording all traffic traversing through the network, starting at midnight (00:00 GMT) on April 1, 2024, and continuing until April 14, 2024, in a pcap file format.

B. Processing of Raw Traffic Dataset

This section outlines the data processing pipeline used to extract and analyze flows from the CWN traffic dataset, as illustrated in Figure 2 and Table I. The goal is to convert raw PCAP files into clean, structured CSV format suitable for both human interpretation and machine analysis, while removing duplicate and redundant entries. A flow is defined as a sequence of packets sharing common attributes: source and destination IPs, ports, protocol type, and direction based on TCP/UDP transport protocols. The process begins with [1]pkt2flow, an open-source tool that splits the raw PCAP files into individual flow-level PCAPs, each representing a unique TCP/UDP flow identified by a 5-tuple (source IP, destination IP, source port, destination port, protocol). Next, each flow is classified through packet labeling. To identify the underlying service or application, nDPI, an open-source Deep Packet Inspection library, is used to inspect packet payloads and assign accurate protocol/application labels. This systematic transformation enables flow-level analysis for traffic characterization [30][31]. As a result, all packets within a specific flow inherit the label assigned to that flow, ensuring consistent categorization across the dataset. The unformatted .csv files obtained were further subjected to cleansing using the custom Python script, clean_label-csv.py. This process eliminates duplicates and extraneous files that may have occurred from the repetition of identical packets, overlapping sessions, or excessive retransmissions due to poor link quality. In this case, TCP flows with fewer than 19 packets and UDP flows with fewer than 14 packets were excluded to eliminate incomplete or insignificant flows. TCP requires at least 7 packets for connection setup (ACK, SYN-ACK, ACK) and teardown (FIN, ACK), with additional packets emanating from segmented payloads based on the MSS (1460 bytes) and corresponding acknowledgments. Hence, flows with fewer than 19 packets are considered incomplete or have sufficient data that merit analysis. Unlike TCP, which is connection-oriented, UDP functions without establishing a connection, thus bypassing the necessity for a handshake or disconnection procedures. However, transient or minor data flows, such as those generated by error messages, DNS queries, or attempted connections, usually consist of a smaller number of packets [32][33]. For this reason, a threshold of at least 14 packets is used to determine flows with meaningful information for analysis. Consequently, we improved the dataset's quality by reducing its size by 55%. Following this, the data is structured into a tabular format, where rows indicate instances and columns reflect the features of the dataset. The Table 1 below shows a sample summary of the processed dataset/flows. For privacy and ethical reasons, we have masked the IP addresses.

[1]https://github.com/caesar0301/pkt2flow: cross-platform utility to classify packets into flows.

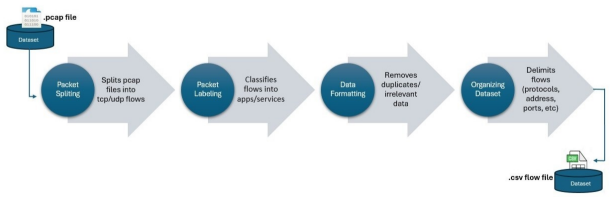

Fig. 2: Data Processing Pipeline Activities

TABLE I: Sample Flows of Processed Dataset

SrcAddr	SrcPort	DstAddr	DstPort	Prot	Label	Pkt	Bytes
***	42857	***	63832	udp	BitTorrent	58	5652
***	53764	***	443	tcp	GoogleServices	19	6050
***	55338	***	443	udp	YouTube	35	13720
***	47566	***	443	tcp	TLS	22	5682
***	38659	***	443	tcp	FbookReelStory	35	8937

C. Application Flow Classification

A total of 88 distinct applications were identified, generating 1,181,765 flows, 348 million packets, and over 287 GB of traffic. For systematic analysis, flows were grouped into eight categories based on functionality and traffic behavior. These include: Gaming (low-latency traffic from platforms like RiotGames and Xbox), P2P File-Transfer (decentralized file-sharing), Social Media (interactive platforms like Facebook and Instagram), System & Software Updates (OS and application patch traffic, e.g., Windows Update), Video Streaming (services like YouTube and Netflix), VoIP/Audio Streaming (real-time voice and music, e.g., Zoom and Spotify), and Web Services & Cloud Applications (browser-based and SaaS traffic). Traffic not aligning to any of the categories, such as locally developed apps or privacy-oriented services like ADS-Analytics and Tor, is grouped under Other Services. Each category encapsulates a specific class of network activity, aiding in the analysis of application-level trends and evolving patterns. This process involves two main activities: (1) Protocol/Service Separation, and (2) Application Categorization, each designed to isolate meaningful user-level application flows from background or encapsulating protocols such as TLS, QUIC, STUN, among others.

1) Protocol/Service Separation: In order to differentiate between actual application traffic and lower-layer protocol overhead, a filtering mechanism was implemented using the Python library pandas and nDPI (deep packet inspection). First, we identified and mapped all protocols/services flows based on their nDPI labels from the labeled dataset. These included protocols such as TLS, SMTP, STUN, QUIC, etc, which are commonly used for session establishment, encryption, or transport but do not themselves represent end-user applications. This approach ensured that traffic attributed to application-layer usage was analyzed independently of encryption or transport-layer encapsulation, reducing protocol-induced noise in the results.

2) Application Categorization: In order to differentiate between actual application traffic and After separating application flows, we proceeded to classify each identified application flow into one of eight (8) distinct categories. This classification was performed using a custom Python function technique that leverages nDPI-assigned labels and well-known destination port numbers [34]. We created a Python function to automate the assignment of each flow to a category. This technique effectively isolates application-layer traffic from protocol-layer noise such as TLS or QUIC encapsulation, allowing for more accurate measurements of user-level behavior and bandwidth consumption. By relying on well-defined rules derived from nDPI labels and known port numbers, the methodology provides a deterministic and transparent classification strategy that can be easily audited and refined. Furthermore, the classification function can be extended to include additional applications or ports, making it adaptable for diverse network environments or evolving traffic profiles. Finally, by leveraging both protocol filtering and rule-based classification, our devised technique ensures scalable and interpretable network traffic categorization suitable for real-world deployment.

IV. Analysis

To analyze traffic behavior and identify usage patterns, we leveraged the application categories using key network metrics: unique application counts, flow counts, packet volumes, and bytes per flow. This application-centric characterization enables the identification and analysis of traffic behaviors, patterns, and attributes unique to specific applications. Flow count is denoted as the number of distinct flows generated by an application within a given time frame. On the other hand, the number of packets refers to the total packets transmitted per flow, while the byte quantifies the volume of data (in bytes) transferred by each application. In this analysis, we considered a flow as a unidirectional sequence of packets sharing the same source/destination IP, source/destination port, and protocol. Flows were extracted and aggregated per application using pkt2flow and nDPI, as detailed in Section III-B, thereby facilitating the computation and comparison of variations in network application traffic. Subsequently, we examined the contributions, distributions, variations and correlations among applications using the previously defined metrics, and evaluated evolving trends across multiple temporal resolutions, including biweekly, weekly, daily, day-night cycles, and hourly intervals. Our technique employed Python's Pandas library to perform data grouping, pivoting, summation and compute time-series statistics for each time scale.

V. Result and Discussion

In this section, we present the results of our empirical analysis and discuss their implications for the network. Our findings are examined across the different temporal resolutions including biweekly, weekly, daily, day-night cycles, and hourly intervals to highlight evolving trends.

TABLE II: Categories of Application Flows

	App_Category	No. of Application	No. of Flows	No. of Packets	No. of Bytes(Gb)
1	Gaming Application	8	1160	128618.0	6.788069e+07
2	Other Services	5	35009	13027795.0	1.015754e+10
3	P2P File-Transfer	1	19018	32965577.0	2.197138e+10
4	Social Media	17	85860	20593668.0	1.335094e+10
5	System & Software Update	10	9577	1791004.0	1.259147e+09
6	Video Streaming	8	198456	217979396.0	1.970275e+11
7	VoIP/Audio Streaming	10	2879	3374241.0	2.276589e+09
8	Web Service & Cloud Application	29	829806	58776869.0	4.092212e+10

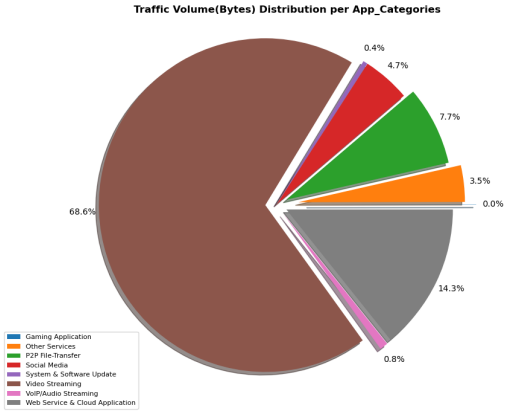

Fig. 3: Traffic Volume(bytes) distribution per App_Categories

A. Bi-Weekly Traffic Trend

This analysis examines the full two-week dataset, a total of 88 distinct applications were identified, as depicted in Table II. Web Services and Cloud Applications accounted for the largest share, comprising approximately 33% of the total applications used in the network. Social media followed, encompassing 17 unique application types. Both Video Streaming and System & Software Update categories ranked third, each comprising 10 application types. The File-Transfer category was the least represented, with only one application type observed. Web Services & Cloud Applications emerged as the most frequently accessed category, with 829,806 flow records across 29 apps. However, their traffic volume (40.92 GB) was relatively moderate, likely due to lighter payloads typical of web browsing, document editing, and SaaS usage. Video Streaming, while comprising only 8 applications, generated the highest volume of traffic, with 197.03 GB, approximately 69% and over 217 million packets, underscoring its bandwidth-intensive nature. Social Media apps were the second most diverse (17 apps) and contributed 85,860 flows and 13.35 GB of traffic, a clear reflection of modern content sharing and interaction habits. P2P File Transfer, represented by a single application, resulted in a massive 21.97 GB of traffic from just 19,018 flows, demonstrating large data exchanges per session typical of torrent-based systems. VoIP/Audio Streaming and Gaming Applications recorded low flow counts and traffic volumes, suggesting their reliance on low-latency, small-packet exchanges, often over UDP or QUIC protocols. Other Services included traffic from less-categorized or locally developed applications, amounting to over 10 GB, which warrants further classification using ML-based analysis. The share distribution of these application categories in terms of traffic volume generation is illustrated with a pie chart in Figure 3. The observed trend highlights the heterogeneity in application behavior within a community network. While Web Services dominate in access frequency, Video Streaming leads in traffic volume, and P2P and Other Services contribute disproportionately to total data transfer. The findings suggest a need for adaptive QoS policies, bandwidth allocation, and traffic shaping strategies.

B. Weekly Traffic Trend

This analysis observes traffic behavior on a week-by-week basis, highlighting changes in usage trends. As shown in Table III and Figure 4(b), Gaming is notably the only application category that demonstrates consistent trend across all metrics, with increases of 5.58% in flows, 3.33% in packet count, and 6.44% in traffic volume across weeks. This is likely increased in user engagement or the growing popularity of specific platforms. Conversely, System and Software Updates experienced the sharpest declines, with drops in flows (0.41%), packets (0.57%), and traffic (0.61%), despite the increase in 2 more applications. This could indicate improved update efficiency or scheduled activity during off-peak hours. Web Service and Cloud Applications, the largest traffic category, saw moderate decreases across all metrics, possibly reflecting reduced enterprise use or improved optimization. VoIP and Audio Streaming showed a unique pattern, increasing both packet and traffic volume despite a reduction in application count, pointing to improved traffic efficiency. Furthermore, Gaming is the only category with significant positive change, while others show slight declines. VoIP's traffic gains with fewer applications suggest a focus on high-usage or higher-quality services. Deeper insight in reference to Figure 4(a) confirms that flows, packets, and traffic typically move together across categories, indicating stable usage patterns. Interesting to note are some observed anomalies, such as the inverse relationship between application count and traffic in System Updates, and the strong growth of Gaming without a corresponding rise in application number, Figure 4(a). These findings offer actionable insights for network administrators. They help identify growth areas, highlight inefficiencies, and guide capacity planning and QoS policy decisions. Overall, the weekly comparison reflects the evolving nature of network demand and emphasizes the importance of application-specific analysis for adaptive network management.

C. Daily Traffic Trend

Figure 5 shows the daily distribution of traffic and flow patterns across application categories. Traffic volumes for application categories are represented by bars, flow counts by bubbles, and the KDE curve emphasizes the smoothed trend of average weekly traffic. Peer-to-peer (P2P) file-transfer traffic exhibited sporadic yet intense behavior in the daily analysis.

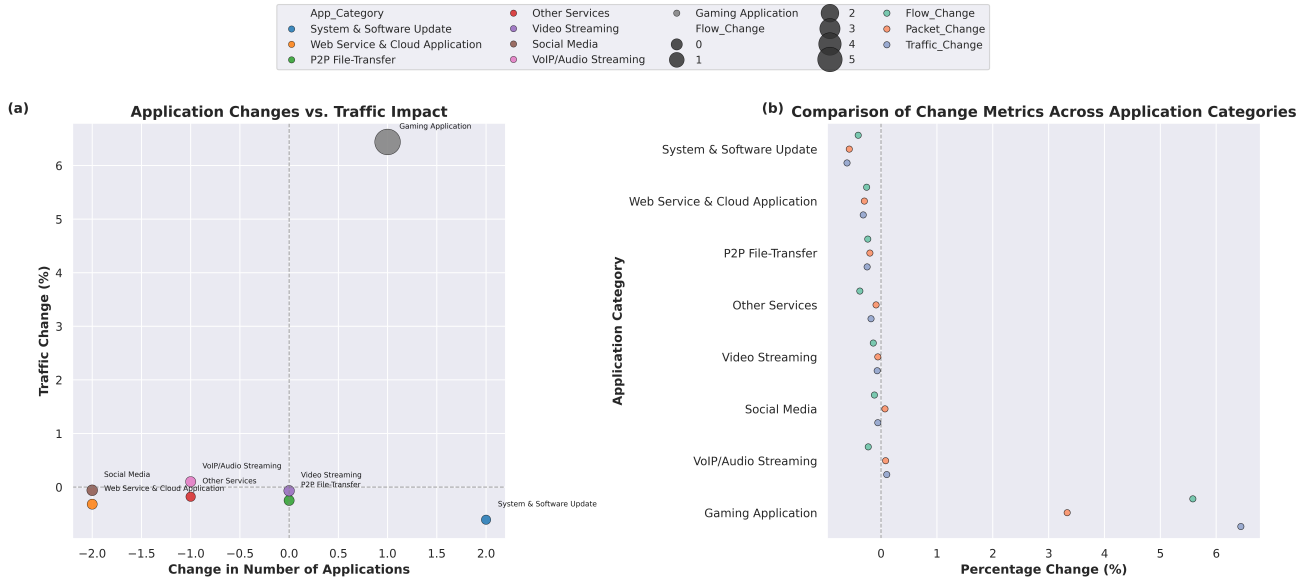

Fig. 4: Week-by-Week traffic analysis and trend

TABLE III: Summary of Weekly Traffic Observation

	App_Category	Change in Apps	Flow Change (%)	Packet Change (%)	Traffic Change (%)
1	Gaming Application	1	5.58	3.33	6.44
2	Other Services	-1	-0.38	-0.09	-0.18
3	P2P File-Transfer	0	-0.24	-0.2	-0.25
4	Social Media	-2	-0.12	0.07	-0.06
5	System & Software Update	2	-0.41	-0.57	-0.61
6	Video Streaming	0	-0.14	-0.06	-0.07
7	VoIP/Audio Streaming	-1	-0.23	0.08	0.1
8	Web Service & Cloud Application	-2	-0.26	-0.3	-0.32

Monday traffic peaked at 8.5 GB but dropped to 272 MB by Wednesday. Extremely high packet-per-flow ratios, such as 2,572, suggest large file exchanges. A Tuesday anomaly 264 flows generating over 1.0 GB underscores the bursty and unpredictable nature of P2P, warranting careful monitoring and potential throttling. Meanwhile, Social media traffic remained consistently high, with daily flows between 10,000 and 15,600 and bytes-per-flow averaging 135–165 KB. Its steady use across weekdays and weekends, including 1.6 GB on Saturday, reflects both commercial and personal engagement. Gaming traffic was distinctly weekend-focused. Flows jumped from a weekday low of 42 to 234 on Saturday, while Sunday traffic reached 25 MB, four times Monday's volume. High bytes-per-packet values (1.8 KB) point to rich content or new updates. This temporal concentration suggests a need for targeted bandwidth allocation during off-peak hours. VoIP/Audio streaming, though low in volume, exhibited consistent, latency-sensitive behavior, with 300–632 daily flows and roughly 0.75 KB per packet, and peaking at 828MB on Monday before declining over the weekend. These applications require prioritization during work hours to maintain quality. System and software updates followed a weekday-centric pattern, with Monday traffic reaching 528MB, reflecting structured enterprise patching. Other services, likely related to backups or syncs, maintained steady flow counts and peaked at 2.1GB on Monday, contributing to persistent background load. A day-of-week breakdown revealed Monday as the busiest day across most categories, especially business-related services. By Friday, business traffic declined as gaming increased, leading to weekend surges dominated by entertainment. Sunday marked the week's traffic peak, totalling 40.4TB, with video streaming alone accounting for around 90% of bandwidth. Though less frequent, P2P traffic caused sudden load spikes, posing risks to network stability. Weekday traffic was dominated by web services and VoIP, while evenings and weekends shifted toward gaming and video streaming.

Temporal analysis revealed distinct usage behaviors between weekdays and weekends. On weekdays, both web services and VoIP traffic volumes peaked during standard business hours, aligning with enterprise and communication demands. In contrast, gaming traffic experienced a significant surge during weekends, with usage on Sunday rising to 81 times the volume observed on Monday. This trend reflects increased recreational use during non-working days. Outliers such as P2P's Monday surge and gaming's 81-fold traffic increase on Sunday highlight the need for proactive detection. Overall, the daily traffic analysis illustrates how application types and usage timing shape network demand. Adaptive strategies, such as time-aware bandwidth provisioning, QoS enforcement, and anomaly detection, are essential.

D. Hourly Traffic Trends

Analysis of hourly traffic shows peak usage periods, and which applications drive the highest demand, as shown in Figure 6. As demonstrated in Figure 6(b), Web Services and Cloud Applications consistently account for the highest volumes of flows, packets, and bytes, driven by enterprise activity

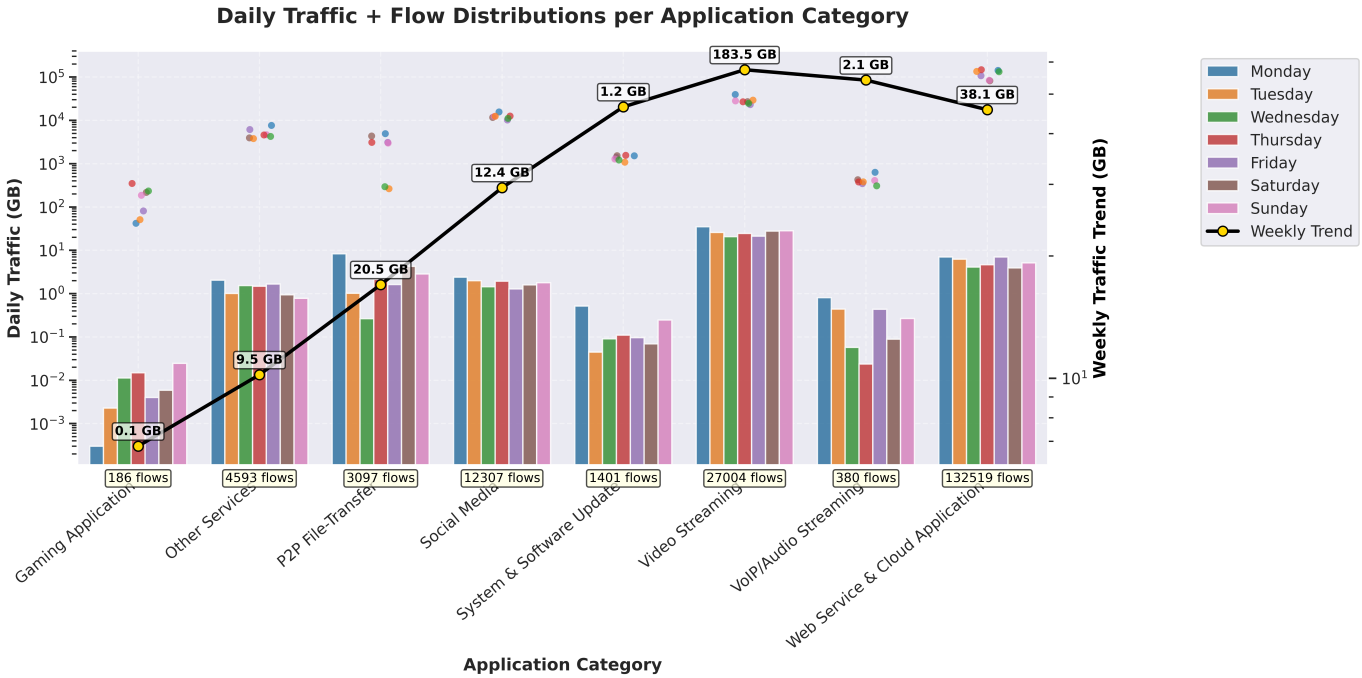

Fig. 5: Daily application traffic analysis with average weekly traffic usage

during standard business hours. In contrast, Video Streaming, while generating fewer flows, dominates total byte volume due to its bandwidth-intensive nature, particularly during evening peaks. Social media maintains steady engagement throughout the day, contributing significantly across all metrics. P2P File-Transfer applications, though less frequent, produce exceptionally high per-session byte volumes, reflecting heavy but infrequent transfers. Temporal trends highlight a clear divide between business and leisure usage. Web services sustain steady throughput across the day, aligning with business operational hours. In contrast, entertainment-related applications such as video streaming and gaming spike between 18:00 and 22:00, especially on weekends. Gaming traffic, in particular, intensifies on Saturdays and Sundays, reinforcing the network's dual-use profile. Hourly data further emphasizes this shift: web services dominate the 09:00–17:00 window, while entertainment traffic rises sharply in the evening. Between midnight and 06:00, user-driven traffic declines, giving way to background activity such as P2P transfers and automated updates, Figure 6(a).

E. Irregularities Analysis Trend

Some identified anomalies in the dataset warrant attention. Notably, in Video Streaming, P2P File-Transfer, and Gaming traffic, a possible shift in user behaviour, automated activity, or external interference is observed, as illustrated in Figure 7. Video streaming, which typically peaks in residential evening hours (6–11 PM), exhibited irregular daytime spikes and nocturnal surges, possibly indicating automated processes. P2P traffic, usually stable with overnight peaks, showed unusual daytime surges, potentially linked to coordinated file-sharing

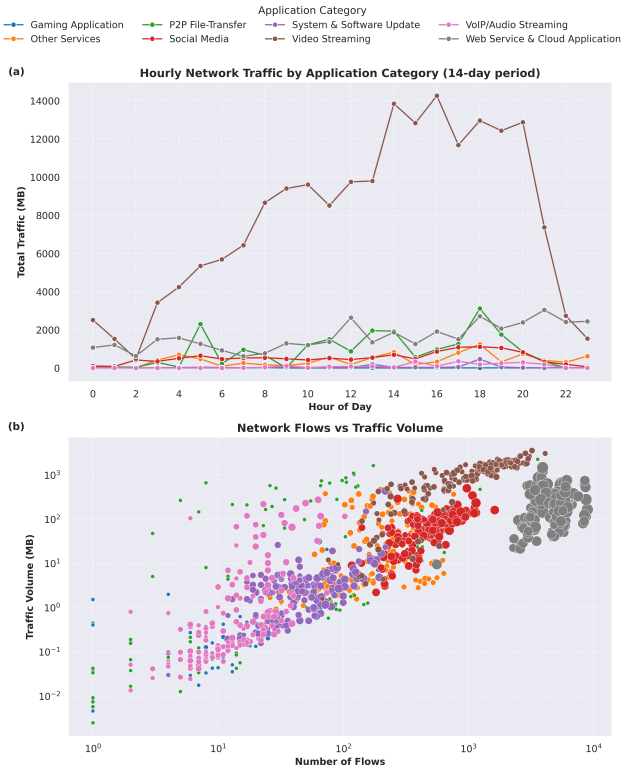

Fig. 6: Hourly analysis of the iNethi CWN traffic and trend

or throttling. Meanwhile, Gaming traffic, expected to rise in evenings and weekends, displayed anomalous early-morning spikes and weak weekend activity, suggesting non-human

usage or service disruptions. These deviations highlight potential behavioral or technical abnormalities requiring further investigation to assess their causes and network impact.

The hourly timescale analysis clearly shows that network usage is shaped by both application type and time. As observed in Figures 5 and 6, business applications dominate weekdays and working hours, while entertainment traffic peaks during off-hours and weekends. The revealed patterns provide actionable intelligence for improving network efficiency, securing resources, and ensuring a quality end-user experience.

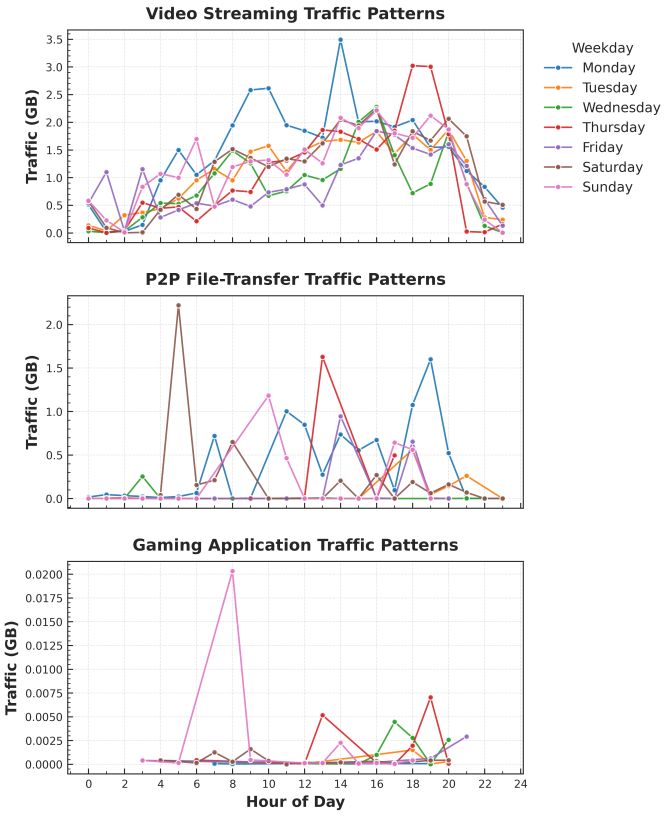

Fig. 7: Analysis of irregularities and patterns

VI. Conclusion and Recommendation

In accordance with our objective, the findings reveal distinctive insights into the traffic dynamics within Community Wireless Networks (CWNs), revealing notable heterogeneity, with both application type and time shaping the network usage. Web Services dominate usage frequency, while Video Streaming consumes the highest bandwidth, underscoring the need for adaptive Quality of Service (QoS) and prioritized bandwidth allocation in these settings. Temporal analysis highlights distinct usage patterns: business-oriented applications such as Web Services and VoIP peak during weekday work hours, while recreational services like Gaming exhibit dramatic weekend surges, with Sunday traffic volumes up to 81 times higher than Monday. Furthermore, the analysis reveals weekly declines in most application usage, except Gaming, signalling behavioral shifts or infrastructure changes. High correlation between flow, packet, and byte metrics supports data-driven capacity planning. Interestingly, identified outliers, such as anomalous Video Streaming peaks and erratic P2P traffic, unravel possible throttling, non-standard usage, or disruptions, necessitating root-cause analysis. In summary, this analysis underscores the need for time-sensitive, application-aware management in such settings, and at the same time offers operators a guide to optimize efficiency, implement QoS controls, and enhance service quality. Future research should investigate: (1) real-time adaptive traffic control for observed abnormalities, and (2) deep packet inspection to analyze traffic from locally developed applications.

References

[1] S. Sevilla, M. Johnson, P. Kosakanchit, J. Liang, and K. Heimerl, "Experiences: Design, implementation, and deployment of colte, a community lte solution," in *The 25th Annual International Conference on Mobile Computing and Networking*, 2019, pp. 1–16.

[2] A. Cilfone, L. Davoli, L. Belli, and G. Ferrari, "Wireless mesh networking: An iot-oriented perspective survey on relevant technologies," *Future Internet*, vol. 11, 2019.

[3] J. Jiménez, R. Baig, P. Escrich, A. M. Khan, F. Freitag, L. Navarro, E. Pietrosemoli, M. Zennaro, A. H. Payberah, and V. Vlassov, "Supporting cloud deployment in the guifi. net community network," in *Global Information Infrastructure Symposium-GIIS 2013*, 2013, pp. 1–3.

[4] Y. Teng, M. Liu, F. R. Yu, V. C. Leung, M. Song, and Y. Zhang, "Resource allocation for ultra-dense networks: A survey, some research issues and challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2134–2168, 2018.

[5] Y. Sun, Z. Xing, and G. Liu, "Achieving resilient cities using data-driven energy transition: A statistical examination of energy policy effectiveness and community engagement," *Sustainable Cities and Society*, vol. 101, p. 105155, 2024.

[6] J. Tooke and J. Chavula, "Resource-constrained real-time network traffic classification using one-dimensional convolutional neural networks," *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*, vol. 443 LNICST, pp. 107–127, 2022.

[7] S. Bhattacharjee and S. K. Das, "Building a unified data falsification threat landscape for internet of things/cyberphysical systems applications," *Computer*, vol. 56, pp. 20–31, 2023.

[8] A. Mehra, V. Sairam, and K. Mittal, "Blendnet: An assisted digital distribution platform for underserved populations," in *Proceedings of the 7th ACM SIGCAS/SIGCHI Conference on Computing and Sustainable Societies*, 2024, pp. 1–17.

[9] P. Micholia, M. Karaliopoulos, I. Koutsopoulos, L. Navarro, R. B. Vias, D. Boucas, M. Michalis, and P. Antoniadis, "Community networks and sustainability: a survey of perceptions, practices, and proposed solutions," *IEEE Communications Surveys & Tutorials*, vol. 20, pp. 3581–3606, 2018.

[10] P. Frangoudis, G. Polyzos, and V. Kemerlis, "Wireless community networks: an alternative approach for nomadic broadband network access," *IEEE Communications Magazine*, vol. 49, pp. 206–213, 5 2011. [Online]. Available: http://ieeexplore.ieee.org/document/5762819/

[11] B. Braem, C. Blondia, C. Barz, H. Rogge, F. Freitag, L. Navarro, J. Bonicioli, S. Papathanasiou, P. Escrich, R. B. Vinas *et al.*, "A case for research with and on community networks," pp. 68–73, 2013.

[12] A. Phokeer, S. Hadzic, E. Nitschke, A. V. Zyl, D. Johnson, M. Densmore, and J. Chavula, "inethi community network: A first look at local and internet traffic usage," in *Proceedings of the 3rd ACM SIGCAS Conference on Computing and Sustainable Societies*, 2020, pp. 342–344.

[13] A. O. Durahim and E. Savaş, "A-make: An efficient, anonymous and accountable authentication framework for wmns," in *2010 Fifth International Conference on Internet Monitoring and Protection*, 2010, pp. 54–59.

[14] J. Backens, G. Mweemba, and G. V. Stam, "A rural implementation of a 52 node mixed wireless mesh network in macha, zambia," *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering*, vol. 38 LNICST, pp. 32–39, 2010.

[15] R. Baig, R. Roca, L. Navarro, and F. Freitag, "guifi. net: A network infrastructure commons," in *Proceedings of the Seventh International Conference on Information and Communication Technologies and Development*, 2015, pp. 1–4.

[16] L. Gwaka, M. Haseki, and C. S. Yoo, "Community networks as models to address connectivity gaps in underserved communities," *Information Development*, vol. 39, pp. 524–538, 9 2023.

[17] M. A. Lopez, M. Baddeley, W. T. Lunardi, A. Pandey, and J. P. Giacalone, "Towards secure wireless mesh networks for uav swarm connectivity: Current threats, research, and opportunities," *Proceedings - 17th Annual International Conference on Distributed Computing in Sensor Systems, DCOS 2021*, pp. 319–326, 2021.

[18] A. Neumann, L. Navarro, and L. Cerda-Alabern, "Enabling individually entrusted routing security for open and decentralized community networks," *Ad Hoc Networks*, vol. 79, pp. 20–42, 2018.

[19] P. Velan, J. Medková, T. Jirs´, and P. Čeleda, "Network traffic characterisation using flow-based statistics," in *NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium*, 2016, pp. 907–912.

[20] P. Pareek and A. Thakkar, "A survey on video-based human action recognition: recent updates, datasets, challenges, and applications," *Artificial Intelligence Review*, vol. 54, pp. 2259–2322, 2021.

[21] T. Benson, A. Akella, and D. A. Maltz, "Network traffic characteristics of data centers in the wild," in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, 2010, pp. 267–280.

[22] H. Riggs, S. Tufail, I. Parvez, M. Tariq, M. A. Khan, A. Amir, K. V. Vuda, and A. I. Sarwat, "Impact, vulnerabilities, and mitigation strategies for cyber-secure critical infrastructure," *Sensors*, vol. 23, p. 4060, 2023.

[23] K. White, D. L. Johnson, S. Hadzic, and M. Densmore, "Community networks powered by community currencies," in *Proceedings of the 6th ACM SIGCAS/SIGCHI Conference on Computing and Sustainable Societies*, 2023, pp. 120–123.

[24] L. Maccari, "Detecting and mitigating points of failure in community networks: A graph-based approach," *IEEE Transactions on Computational Social Systems*, vol. 6, pp. 103–116, 2019.

[25] L. Maccari and R. L. Cigno, "A week in the life of three large wireless community networks," *Ad Hoc Networks*, vol. 24, pp. 175–190, 2015.

[26] M. Fomenkov, K. Keys, D. Moore, and K. C. Claffy, "Longitudinal study of internet traffic in 1998-2003," in *Proceedings of the winter international synposium on Information and communication technologies*, 2004, pp. 1–6.

[27] P. Jurkiewicz, G. Rzym, and P. Boryło, "Flow length and size distributions in campus internet traffic," *Computer Communications*, vol. 167, pp. 15–30, 2021.

[28] O. Malyeyeva, Y. Davydovskyi, and V. Kosenko, "Statistical analysis of data on the traffic intensity of internet networks for the different periods of time," in *CEUR Workshop Proceedings*, 2019, pp. 897–910.

[29] P. Goyal and A. Goyal, "Comparative study of two most popular packet sniffing tools-tcpdump and wireshark," in *2017 9th International Conference on Computational Intelligence and Communication Networks (CICN)*. IEEE, 2017, pp. 77–81.

[30] L. Deri, M. Martinelli, T. Bujlow, and A. Cardigliano, "ndpi: Open-source high-speed deep packet inspection," in *2014 International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2014, pp. 617–622.

[31] B. Lypa, I. Horyn, N. Zagorodna, D. Tymoshchuk, and T. Lechachenko, "Comparison of feature extraction tools for network traffic data," *arXiv preprint arXiv:2501.13004*, 2025.

[32] V. Paxson, "Strategies for sound internet measurement," in *Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement*, 2004, pp. 263–271.

[33] F. Schmidt, M. H. Alizai, I. Aktaş, and K. Wehrle, "Refector: Heuristic header error recovery for error-tolerant transmissions," in *Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies*, 2011, pp. 1–12.

[34] M. Cotton, L. Eggert, J. Touch, M. Westerlund, and S. Cheshire, "Internet assigned numbers authority (iana) procedures for the management of the service name and transport protocol port number registry," Tech. Rep., 8 2011.