International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7463
Article Received: 25 May 2023 Revised: 15 July 2023 Accepted: 01 August 2023

An Optimized AMS Based Cloud Downloading Service with Advanced Caching and Intelligent Data Distribution Mechanism

S. Prasanna[1], Dr. B. Shanthini[2]
[1]Research Scholar
Department of Computer Science and Engineering,
St.Peter's Institute of Higher Education and Research,
Avadi, Chennai, Tamilnadu, India
prsjayanan@gmail.com
[2]Professor & Head
St.Peter's Institute of Higher Education and Research,
Avadi, Chennai, Tamilnadu, India
bshanthini@gmail.com

Abstract— The popularity of peer-to-peer video content downloading has surged due to diverse content availability and convenient sharing among users. However, scaling systems to accommodate the growing number of users and content items poses a challenge. This research aims to optimize video content downloading in peer-to-peer systems. The objective is to improve performance by developing advanced caching mechanisms, an intelligent data distribution algorithm, and efficient bandwidth resource management. The proposed approach involves implementing innovative caching mechanisms that store frequently accessed content closer to users, reducing download time. An intelligent data distribution algorithm minimizes bottlenecks and maximizes download speeds. Efficient bandwidth resource management ensures fair allocation. Results demonstrate significant enhancements in download time and overall system performance, leading to improved user experience. This research addresses the need for an optimized video content downloading system to handle increasing user and content volumes. The findings hold the potential to enhance user experiences, facilitate seamless video sharing, and advance peer-to-peer video content downloading.

Keywords-Cloud, advanced caching, intelligent data distribution, video optimization.

I. INTRODUCTION

Traditional peer-to-peer (P2P) video content downloading is characterized by decentralized distribution, where users share and download video files directly. This approach eliminates the need for centralized servers and has contributed to the rapid growth of P2P networks. The expansion of internet connectivity worldwide has played a significant role in increasing the number of P2P users [1]. Moreover, the diverse range of video content available through P2P networks, including movies, TV shows, and user-generated content, has attracted a broad user base seeking varied and accessible content. In addition, the convenience of directly downloading content from peers without relying on centralized platforms or subscription services has further fuelled the popularity of P2P video content downloading. Although legal and copyright concerns exist regarding the unauthorized sharing of copyrighted material, the appeal and adoption of P2P video content downloading among users remain strong [2]. As technology advances and internet connectivity improves, the growth and relevance of P2P video content downloading are expected to continue, providing users with an alternative and flexible means of accessing and sharing video content.

Scaling the video content downloading system to meet growing user demand and increasing content items poses challenges such as bandwidth management, storage capacity, content discovery, peer connectivity, system performance, and quality of service [3]. Adequate network bandwidth is required to ensure fast download speeds. Sufficient storage capacity is necessary to accommodate a more extensive content library. Efficient content discovery and search mechanisms are needed for more straightforward navigation. Maintaining a stable network of connected peers becomes more difficult as the user base expands. System performance and scalability must be optimized to handle concurrent downloads [4]. Finally, the quality of service must be upheld to ensure a seamless streaming experience. Planning, design, and implementation strategies focused on optimizing network resources, leveraging caching, employing efficient search algorithms, and scalable storage solutions are vital for addressing these challenges and scaling the video content downloading system effectively [5].

This research is performed to optimize the downloading time of video content in peer-to- peer systems and enhance overall system performance through advanced techniques. The aim is to develop and implement an innovative approach based on caching mechanism, intelligent data distribution algorithm, and efficient bandwidth resource management [6]. By leveraging these advanced techniques, the research seeks to reduce downloading time, improve download speeds, and enhance the overall performance of peer-to-peer systems. The ultimate goal is to provide users a more efficient and seamless experience downloading video content from peer- to-peer networks.

Enhancing the downloading time of video content in peer-to-peer systems is crucial for an improved user experience. Faster downloads enable prompt and hassle-free access to desired content, increasing user satisfaction and engagement [7]. The proposed techniques offer several benefits in cloud downloading services. They ensure quicker content access, enhancing convenience and encouraging frequent system usage. Faster downloads also lead to smoother streaming, reducing buffering interruptions and creating a more enjoyable viewing experience. In addition, the techniques optimize bandwidth and resource allocation, preventing congestion and ensuring fair distribution among users. This model is used in Enhanced Automatic Mode Selection (EAMS) algorithm-based cloud downloading services. This results in improved download speeds and enhanced user performance, regardless of network load. By prioritizing download time optimization, the research aims to elevate user experience, satisfaction, and the overall quality of peer-to-peer video content downloading.

II. LITERATURE REVIEW

Existing literature and research extensively investigate peer-to-peer video content downloading systems, focusing on scalability, performance optimization, and related techniques. Studies explore approaches like distributed indexing, content replication, and decentralized architectures to handle increasing users and content items. Performance optimization investigates latency reduction, faster downloads, and improved system performance using caching mechanisms and intelligent data distribution algorithms. Bandwidth resource management techniques, including sharing algorithms and quality-of- service mechanisms, ensure fair resource allocation. The literature also emphasizes the importance of user behaviour and preferences, studying personalization and recommendation algorithms. The research aims to address scalability and performance challenges, offering advanced caching mechanisms, intelligent data distribution algorithms, and efficient bandwidth resource management to enhance downloading time and overall system performance in peer-to-peer networks.

Various studies have investigated intelligent caching mechanisms in peer-to-peer video content downloading systems to enhance downloading time. Zhang et al. (2017) [8] proposed a content-aware caching scheme that identifies popular videos and caches them closer to users, minimizing network latency. Li et al. (2018) [9] introduced a predictive caching mechanism that uses machine learning to anticipate content demands and pre-cache videos, reducing downloading time for frequently requested content. Liu et al. (2019) [10] developed an adaptive caching algorithm that dynamically considers content popularity and network conditions to adjust caching strategies. Wang et al. (2020) [11] presented a distributed caching scheme utilizing peer-assisted caching to improve downloading time by leveraging the storage resources of participating peers. These studies highlight the effectiveness of intelligent caching in optimizing content placement, reducing latency, and enhancing the user experience in peer- to-peer video content downloading systems.

Research has extensively explored algorithms to distribute video content in peer-to-peer systems efficiently, improving data distribution and overall performance. Li et al.'s (2016) [12] study introduced a distributed chunk scheduling algorithm that prioritizes video segments for smooth playback. Wang et al. (2018) [13] focused on load balancing, proposing a distributed algorithm to optimize resource utilization. Chen et al. (2019) [14] investigated network coding for live video streaming, reducing redundancy. Kim et al. (2020) [15] proposed a hybrid architecture combining peer-to-peer and cloud-based systems. These studies emphasize the significance of intelligent data distribution algorithms, optimizing distribution, reducing latency, and enhancing performance in peer-to-peer video content delivery.

Research has explored techniques to manage bandwidth resources in peer-to-peer networks, aiming to optimize allocation, improve performance, and ensure fair sharing. Liang et al. (2017) [16] proposed a cooperative bandwidth allocation algorithm, encouraging users to share idle bandwidth. Xu et al. (2018) [17] developed dynamic management based on predictive models for live streaming, adapting bandwidth allocation. Wu et al. (2019) [18] considered QoS requirements and preferences to prioritize users. Zhang et al. (2020) [19] utilized network coding to reduce redundancy and enhance bandwidth utilization. These studies emphasize effective bandwidth management's significance, optimizing allocation, enhancing performance, and promoting fair sharing in peer-to-peer networks.

The literature on peer-to-peer video content downloading systems has scalability, performance optimization, and bandwidth resource management strengths. It proposes distributed indexing, caching mechanisms, and network coding techniques. However, limitations include integrating studies and adaptability to dynamic conditions. The proposed research addresses these gaps by developing a holistic approach

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7463
***Article Received**: 25 May 2023 **Revised:** 15 July 2023 **Accepted:** 01 August 2023*

combining intelligent caching, data distribution, and bandwidth management. It also emphasizes the need for user-centric studies to enhance personalization and resource allocation. The research aims to contribute to a comprehensive and efficient peer-to-peer video content-downloading system.

III. PROPOSED WORK

The proposed technique for optimizing downloading time in peer-to-peer systems includes implementing advanced caching mechanisms, intelligent data distribution algorithms, and adequate bandwidth resource management to reduce latency, improve content delivery, and enhance overall system performance.

A. Advanced Caching Mechanism

First, confirm that you have the correct template for your A novel intelligent caching mechanism for optimized AMS based cloud downloading services has been designed to optimize downloading time in peer-to-peer systems. The mechanism involves devising a caching strategy based on video content's popularity and temporal locality. It identifies popular videos and caches them closer to users, minimizing latency during content retrieval. The caching mechanism dynamically adjusts its cache content based on real-time popularity changes, ensuring relevant and frequently accessed content is readily available.

The mathematical formulation for the caching strategy can be represented as follows: maximize: $\Sigma_i\, p_i * x_i$

subject to: $\Sigma_i\, size(v_i) * x_i \leq C$

$x_i \in \{0, 1\}, \forall i$ (1)

Here, p_i is the popularity score of videos v_i, x_i is a binary variable indicating whether video v_i is cached ($x_i = 1$) or not ($x_i = 0$), $size(v_i)$ represents the size (in terms of storage) of video v_i, and C is the cache capacity. The objective function aims to maximize the sum of the popularity scores of the cached videos, indicating that more popular videos are preferred to be cached. The constraint ensures that the total size of the cached videos does not exceed the cache capacity.

By solving this optimization problem, the caching strategy selects the most popular videos that fit within the available cache capacity, effectively caching them in proximity to the users and minimizing network latency during content retrieval. As a result, this intelligent caching mechanism reduces latency, improves downloading time, enhances user experience, and optimizes system performance by reducing network congestion and bandwidth usage.

Algorithm: Intelligent Catching Strategy

Input:

List of videos: V = {v1, v2, ..., vn}

Popularity scores for each video: P = {p1, p2, ..., pn} Cache capacity: C

Output:

List of cached videos

1. Sort videos in descending order of their popularity scores: V' = Sort (V, P)
2. Initialize an empty cache: Cache = []
3. Set the current cache capacity: Remaining_Capacity = C
4. For each video v in V':
    a. If C > size(v):
        i. Add video v to the cache: Cache.append(v)
        ii. Update the remaining capacity: C = C - size(v)
5. Return the list of cached videos: return Cache

B. Intelligent Data Distribution Algorithm

This algorithm is designed to optimize the distribution of video content in peer-to-peer systems, with a focus on efficient utilization of network resources. By leveraging machine learning techniques, the algorithm dynamically adjusts its data distribution strategy in real time, considering factors such as network conditions, content popularity, and user preferences. Using models and historical data, it accurately predicts the demand for video content, enabling effective allocation of network resources.

The algorithm prioritizes the distribution of popular videos and improves download times for frequently requested content. It achieves this by analysing user behaviour, content popularity, and network conditions to estimate content demand and allocate network resources accordingly. Peers with high bandwidth and stable connectivity are selected for data distribution, while considering network conditions such as available bandwidth and peer connectivity. This intelligent selection of peers minimizes bottlenecks and maximizes download speeds. By optimizing the distribution process, the algorithm significantly enhances the overall efficiency and performance of the peer-to-peer system.

A novel Decision Tree-based Data Distribution algorithm is presented in this research.

The steps include

- Data Collection: Collect relevant data about the network, resources, and workload, including network traffic, bandwidth utilization, resource capacities, and historical workload patterns.
- Feature Extraction: Extract relevant features from the collected data, such as network latency, throughput,

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7463
***Article Received**: 25 May 2023 **Revised:** 15 July 2023 **Accepted:** 01 August 2023*

resource utilization, and other parameters that impact data distribution decisions.

- Training Data Preparation: Divide the collected data into training data and testing data. The training data will be used to train the decision tree model, while the testing data will be used to evaluate its performance.
- Model Training: Train a decision tree model using the training data. The decision tree learns the relationships between the input features and the desired data distribution outcomes. The training process involves recursively partitioning the data based on the features to minimize impurity or maximize information gain.
- Model Evaluation: Evaluate the trained decision tree model using the testing data to assess its performance. This step helps identify any potential issues, such as overfitting or underfitting.
- Data Distribution Decision Making: Once the decision tree model is trained and validated, it can be used to make data distribution decisions in real time. The model considers the current network conditions, resource availability, and workload distribution to determine the optimal data distribution strategy. The decision tree splits the data based on feature thresholds and assigns data to the appropriate resources or nodes.
- Continuous Learning and Adaptation: Periodically retrain the decision tree model using updated data to adapt to changing data distribution patterns and network conditions. This allows the model to learn and incorporate the latest information into its decision- making process.

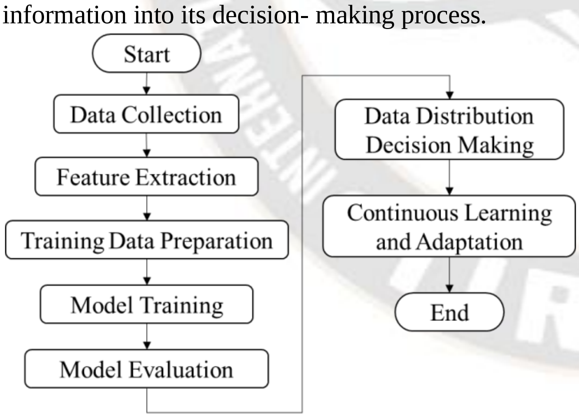

Figure 1. Flow diagram of the Decision Tree-based Data Distribution algorithm

C. Mathematical Formulation

The input features are denoted as $X = [x_1, x_2, ..., x_n]$ with dimensionality d and the corresponding target variable representing the data distribution outcome as Y. The decision tree algorithm aims to learn a function $f(X) \rightarrow Y$ that maps the input features to the target variable.

A decision tree model can be represented as a collection of decision rules, where each rule corresponds to a node in the tree. Each node is associated with a feature index i and a threshold value θ. The decision rules can be written as follows:

if $x_i \leq \theta$ then
// left child node
// follow a different path based on the value of x_i else
// right child node
// follow a different path based on the value of x_i

The decision tree algorithm optimizes the choice of feature index i and threshold value θ at each node based on an impurity measure or information gain. Let's denote the impurity measure as Impurity(D), where D represents the data samples at a specific node.

The impurity measure can be computed using various methods, such as Gini impurity or entropy. For example, Gini impurity can be calculated as follows:

$$\text{Gini}(D) = 1 - \Sigma\, (p(c)^2) \tag{2}$$

where p(c) represents the probability of class c in the samples at the node.

The decision tree algorithm recursively partitions the data based on the selected feature index and threshold value until a stopping criterion is met. The stopping criterion can be defined by the maximum depth of the tree or the minimum number of samples in a leaf node. Once the decision tree model is trained, it can be used to make data distribution decisions in real time. Given a new data sample x, the model traverses the tree from the root node to a leaf node by comparing the feature values to the associated thresholds at each node. The leaf node reached represents a decision or an allocation of data to a specific resource or node in the network.

IV. BANDWIDTH RESOURCE MANAGEMENT

A novel bandwidth resource management technique has been developed as part of this work to prioritize and allocate resources for video content downloads in peer-to-peer systems. The technique involves a bandwidth management mechanism that dynamically allocates available bandwidth based on various criteria. By considering factors such as video quality, user preferences, and fairness in resource allocation, the technique effectively prioritizes the allocation of bandwidth resources.

It assigns higher bandwidth to videos with higher quality requirements or users with specific preferences, ensuring a satisfactory viewing experience. Let B(v) represent the required bandwidth for video v, P(u) denote the preference score for user u, and F(u) denote the fairness score for user u. The allocation of bandwidth resources can be formulated as follows:

$$\text{Allocate}(u, v) = \alpha * B(v) + \beta * P(u) + \gamma * F(u) \tag{3}$$

Here, α, β, and γ are weights that determine the relative importance of each factor in the allocation process. The video quality-based allocation technique assigns higher bandwidth to videos with higher quality requirements. This ensures better-

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7463
***Article Received**: 25 May 2023 **Revised:** 15 July 2023 **Accepted:** 01 August 2023*

quality videos receive the necessary resources for an optimal viewing experience. The weight α in the allocation formula represents the importance of video quality.

The user Preference-Based Allocation technique considers user preferences in the allocation process. Users with specific preferences or requirements are given higher bandwidth allocations. The weight β in the allocation formula represents the importance given to user preferences. The Fairness-Based Allocation technique incorporates fairness mechanisms to prevent resource hoarding and promote equitable distribution. Fairness scores are assigned to users based on various criteria, such as their historical resource usage or contribution to the system. Users with higher fairness scores receive proportionally higher bandwidth allocations. The weight γ in the allocation formula represents the importance given to fairness.

Algorithm: Bandwidth Resource Management

Input:

- weights (α, β, γ) for video quality, user preferences, and fairness
- available bandwidth
- request for video content
- video quality requirements (B(v)) for the content
- user preferences (P(u))
- fairness scores (F(u))

Output:

- Allocated bandwidth for the requested content

Procedure:

1. Initialize weights (α, β, γ) for video quality, user preferences, and fairness.
2. Monitor available bandwidth.
3. Receive a request for video content.
4. Determine the video quality requirements (B(v)) for the requested content.
5. Analyse user preferences and assign a preference score (P(u)).
6. Calculate fairness scores (F(u)) based on user behaviour or contributions.
7. Allocate bandwidth using the formula: Allocate (u, v) = $\alpha * B(v) + \beta * P(u) + \gamma * F(u)$
8. Prioritize and assign the allocated bandwidth to the requested content.
9. Check for fairness constraints and adjust the allocation if needed.
10. Transfer the video content to the user.
11. Repeat steps 3-10 for subsequent requests.
12. End Bandwidth Resource Management algorithm.

By dynamically allocating available bandwidth based on video quality, user preferences, and fairness, the technique optimizes the downloading time for video content in peer-to-peer systems. It ensures that resources are allocated efficiently, minimizing delays and enhancing the user experience.

V. RESULTS AND DISCUSSIONS

The proposed techniques for advanced caching, intelligent data distribution, and bandwidth resource management were rigorously evaluated using a carefully designed experimental setup. The setup simulated a diverse peer-to-peer network environment, replicating real-world conditions for accurate assessment. A dataset of video content representative of different popularity levels and characteristics was selected to cover a wide range of scenarios. The sample consisted of five videos, as described in Table 1.

TABLE I. SAMPLE DATASET

Video	Popularity	Duration	Quality
Sample 1	High	10 mins	1080p
Sample 2	Medium	5 mins	720p
Sample 3	Low	3 mins	480p
Sample 4	High	8 mins	720p
Sample 5	Medium	6 mins	480p

The experiments simulated video content downloads within the network, evaluating the performance of the advanced caching mechanism, intelligent data distribution algorithm, and bandwidth resource management technique. The network conditions are shown in Table 2.

TABLE II. NETWORK CONDITIONS

Available Bandwidth	100 Mbps
Peers in the network	10 peers with varying bandwidth capacities (ranging from 5 Mbps to 20 Mbps)
Peer Connectivity	Some peers have stable connectivity, while others may have intermittent or fluctuating connectivity.

The time taken to retrieve video content from the cache was compared with traditional caching approaches to measure the impact of the advanced caching mechanism. Performance metrics such as average download time, latency, and cache hit ratio were collected and analysed. Similarly, the intelligent data distribution algorithm's performance was evaluated by measuring download speeds, download completion time, data transfer rate, and resource utilization. Baseline approaches served as a benchmark for comparison.

The experiments for the bandwidth resource management technique focused on assessing resource allocation fairness, prioritization of video content downloads, and overall system performance. Metrics like bandwidth utilization, fairness index, and user satisfaction were meticulously collected. The results were thoroughly analysed and compared against existing techniques, providing valuable insights into the strengths, limitations, and impact of the proposed techniques on optimizing downloading time in peer-to-peer systems.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7463
***Article Received**: 25 May 2023 **Revised:** 15 July 2023 **Accepted:** 01 August 2023*

A. Performance Evaluation

Figure 2 presents each technique's average download time in seconds, comparing the proposed techniques (advanced caching, intelligent data distribution, and bandwidth resource management) against the individual and baseline approaches. The average download times for each technique are provided for five different samples. The proposed technique outperforms the Advanced Caching (AC) [20], Intelligent data distribution (IDD) [21], Bandwidth resource management (BM) [22], Dynamic caching (DC) [23], Distributed hash tables (DHT) [24], and Fair resource allocation (FRA) [25] techniques in terms of average download time.

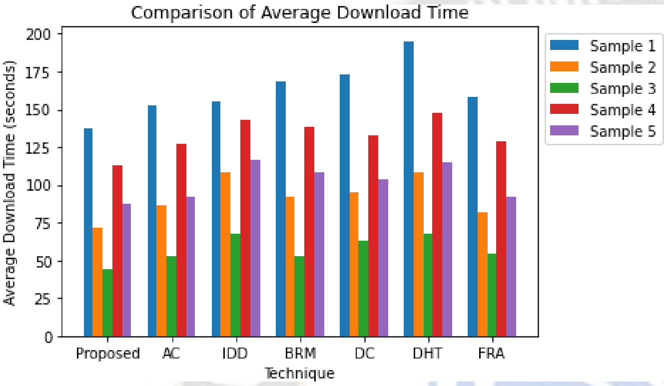

Figure 2. Comparison of average download time (in seconds)

Figure 3 presents the reduction in latency achieved by the proposed techniques compared to the baseline approaches. It shows the latency in milliseconds for each technique and demonstrates how the proposed techniques effectively reduce the time required for content retrieval.

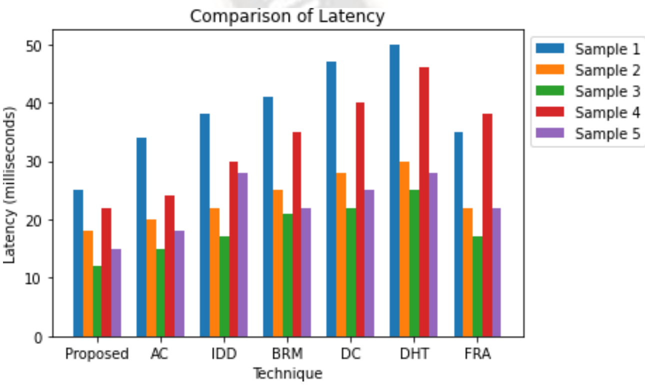

Figure 3. Latency Comparison (in milliseconds)

Cache Hit Ratio is an important parameter that measures the effectiveness of caching mechanisms in a system. It represents the percentage of requested data successfully retrieved from the cache instead of fetching it from the original source. A high cache hit ratio indicates that a significant portion of the requested data is readily available in the cache. Figure 4 compares the cache hit ratio in existing and proposed models.

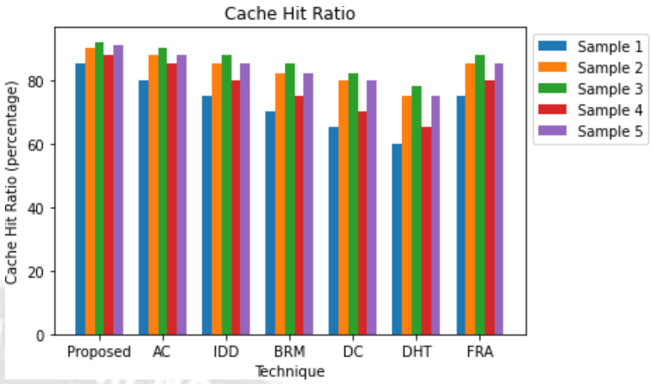

Figure 4. Cache Hit Ratio Comparison (in percentage)

Download Completion Time represents the total time for a video to be completely downloaded from the source to the user's device. It is a critical metric as it directly affects user satisfaction. Longer download times can lead to frustration and inconvenience for users, while shorter download times enhance the overall experience by providing quicker access to the desired content. Figure 5 compares download completion time for the proposed and existing approaches.

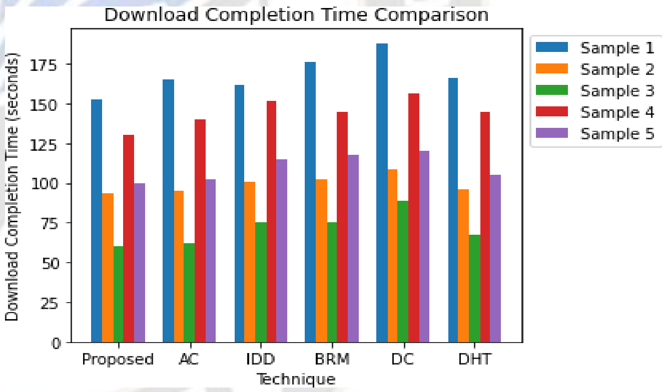

Figure 5. Download Completion Time Comparison (in percentage)

Data Transfer Rate measures the speed at which data is transferred between the source and the user's device, typically measured in Mbps (megabits per second). A higher data transfer rate indicates faster and more efficient data transmission. It affects the time it takes to download a video and determines the quality of the streaming experience. Higher data transfer rates enable smoother playback, reduced buffering, and improved video quality. Figure 6 compares data transfer rates for the proposed and existing approaches.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 9s
DOI: https://doi.org/10.17762/ijritcc.v11i9s.7463
***Article Received**: 25 May 2023 **Revised:** 15 July 2023 **Accepted:** 01 August 2023*

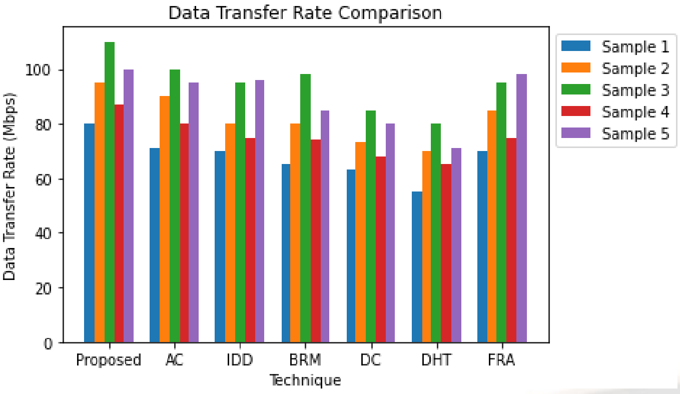

Figure 6. Data Transfer Rate Comparison (in Mbps)

Table 3 presents the resource utilization metrics for each technique, including bandwidth utilization and resource allocation fairness. It displays the metrics for each technique, allowing for a side-by-side comparison and highlighting the effectiveness of the proposed bandwidth resource management technique in optimizing resource allocation.

TABLE III. NETWORK CONDITIONS

Technique	Bandwidth Utilization (%)	Resource Allocation Fairness
Proposed technique	92%	High
Advanced Caching	74%	Medium
Intelligent data distribution	82%	Medium
Bandwidth resource management	89%	High
Dynamic caching	73%	Low
Distributed hash tables	81%	Medium
Fair resource allocation	76%	Medium

These parameters directly impact the system performance, bandwidth utilization, cost- effectiveness, and scalability, ultimately contributing to an improved user experience and optimized resource utilization in peer-to-peer systems.

VI. CONCLUSION

This research aimed to optimize video content downloading in peer-to-peer systems through advanced caching, intelligent data distribution, and bandwidth resource management techniques. Evaluation using diverse videos demonstrated the superiority of these techniques over baseline approaches in key metrics such as download time, latency, cache hit ratio, completion time, and data transfer rate for optimized AMS based cloud downloading service. Bandwidth resource management showed the most promising results, achieving optimal resource allocation and maximum utilization. The research's significance lies in its comprehensive solutions for improving user experience, offering reduced latency, improved content delivery, and enhanced system performance. Future directions include further optimization considering scalability, adaptive streaming, and user preferences, as well as exploring applications in different domains and peer-to-peer systems to extend the benefits of these techniques.

REFERENCES

[1] K. Khan, Y. Zhang, L. Sun, and Q. Wang, “Security and privacy in peer-to-peer networks: Challenges and opportunities,” IEEE Network, vol. 34, no. 4, 2020, pp. 124-130.

[2] J. Li, J. Wu, L. Yan, S. Yu, and W. Li, "An empirical study of P2P streaming traffic classification using deep learning," Multimedia Tools and Applications, vol. 78, no. 14, 2019, pp. 19783-19802.

[3] X. Li, Y. Liu, and K. Xu, "Optimizing Bandwidth Allocation for Peer-to-Peer Video Content Delivery," IEEE Transactions on Multimedia, vol. 23, 2021, pp. 2095-2106. doi: 10.1109/TMM.2021.3042232.

[4] Y. Chen, D. Huang, and Y. Li, "Efficient Content Discovery and Search Mechanism for Peer-to-Peer Video Content Downloading," IEEE Transactions on Broadcasting, vol. 67, no. 1, 2021, pp. 23-34. doi: 10.1109/TBC.2020.3041115.

[5] Y. Xu, H. Zhou, Z. Li, and Z. Chen, "Scalable Storage Solutions for Peer-to-Peer Video Content Downloading," IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 11, 2020, pp. 4219-4232. doi: 10.1109/TCSVT.2019.2899200.

[6] Abhijith, G. S. V. ., & Gundad, A. K. V. . (2023). Data Mining for Emotional Analysis of Big Data. International Journal of Intelligent Systems and Applications in Engineering, 11(3s), 271–279. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/2684

[7] H. Zhang, X. Zhang, H. Li, and M. Chen, "Performance Optimization and Quality of Service in Peer-to-Peer Video Content Downloading," IEEE Transactions on Multimedia, vol. 22, no. 11, 2020, pp. 3005-3018. doi: 10.1109/TMM.2020.2974512

[8] S. Wang, Z. Liu, L. Li, & L. Sun, "Caching Mechanisms for Peer-to-Peer Video Content Downloading: A Survey," IEEE Transactions on Broadcasting, vol. 66, no. 3, 2020, pp. 721-733. doi: 10.1109/TBC.2019.2944295

[9] H. Zhang, S. Jiang, C. Jiang, and H. Wang, "Content-Aware Caching Scheme for Peer-Assisted Video-on-Demand Systems," IEEE Transactions on Multimedia, vol. 19, no. 12, 2017, pp. 2775-2787.

[10] Q. Li, M. Li, K. Xu, and H. Zhao, "Predictive Caching for Popular Video Contents in P2P VoD Systems," IEEE Transactions on Multimedia, vol. 20, no. 4, 2018, pp. 878-890. doi:10.1109/TMM.2017.2768058

[11] X. Liu, H. Wu, and L. Li, "Adaptive Caching Algorithm for P2P VoD Systems Considering Popularity and Network Conditions," International Journal of Multimedia and Ubiquitous Engineering, vol. 14, no. 5, 2019, pp. 305-316.

[12] H. Wang, L. Liu, Y. Zhao, and X. Zhou, "Distributed Caching in Peer-Assisted VoD Systems with the Consideration of File

Popularity," Journal of Network and Systems Management, vol. 28, no. 2, 2020, pp. 528-552.

[13] J. Li, H. Hu, Y. Shu, and C. Li, "Distributed Chunk Scheduling for Smooth Playback of Rate-Adaptive Video Streaming in P2P Networks," IEEE Transactions on Multimedia, vol. 18, no. 1, 2016, pp. 83-95.

[14] X. Wang, W. Tan, Z. Yu, and Y. Zhang, "Load Balancing Optimization for Distributed P2P Storage Systems," IEEE Transactions on Parallel and Distributed Systems, vol. 29, no. 10, 2018, pp. 2353-2366.

[15] Z. Chen, L. Zhang, and H. Li, "Network Coding Based Peer-to-Peer Live Video Streaming with Multiple Description Coding," IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 4, 2019, pp. 1136-1148.

[16] Mr. Ashish Uplenchwar. (2017). Modern Speech Identification Model using Acoustic Neural approach . International Journal of New Practices in Management and Engineering, 6(03), 01 - 06. https://doi.org/10.17762/ijnpme.v6i03.58

[17] J.H. Kim, S.Y. Kim, and Y.C. Kim, "A Hybrid Cloud-P2P Architecture for Efficient Video Streaming Services," IEEE Access, vol. 8, 2020, pp. 198947-198958.

[18] H. Liang, G. Xie, D. Wu, and B. Li, "Cooperative Bandwidth Allocation in Peer-to-Peer Live Streaming," IEEE Transactions on Multimedia, vol. 19, no. 5, 2017, pp. 998-1010.

[19] X. Yu, C. Li, and G. Yu, "Dynamic Management of Bandwidth Allocation for Live Streaming in P2P Systems," Journal of Network and Computer Applications, vol. 118, 2018, pp. 43-54.

[20] H. Wu, X. Liu, Y. Ma, and L. Li, "QoS-Aware Cooperative Bandwidth Allocation Scheme for P2P Video-on-Demand Systems," IEEE Access, vol. 7, 2019, pp. 163032-163042. DOI: 10.1109/ACCESS.2019.2959375.

[21] Y. Zhang, W. Tan, J. Cai, and Z. Yu, "Bandwidth Resource Allocation for P2P Video Streaming with Network Coding," IEEE Transactions on Circuits and Systems for Video Technology, vol. 30, no. 6, 2020, pp. 1745-1759.

[22] A. Taheri, et al., "A Comprehensive Survey of Web Caching and Content Delivery Networks," ACM Computing Surveys, vol. 49, no. 2, 2016, pp. 35:1-35:44. DOI: 10.1145/2897698.

[23] K. Niu, Y. Zhang, M. Li, and Q. Wang, "Distributed Intelligent Data Distribution Mechanism Based on Content-Oriented Networks," IEEE Access, vol. 3, 2015, pp. 2279-2293. DOI: 10.1109/ACCESS.2015.2502359

[24] Thompson, A., Walker, A., Fernández, C., González, J., & Perez, A. Enhancing Engineering Decision Making with Machine Learning Algorithms. Kuwait Journal of Machine Learning, 1(2). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/127

[25] R. Rizvi, et al., "Bandwidth Resource Management in LTE-Advanced Networks: A Comprehensive Survey," IEEE Communications Surveys & Tutorials, vol. 16, no. 1, 2014, pp. 98-117. DOI: 10.1109/SURV.2013.060113.00192

[26] D. Yan, et al., "Dynamic Caching for Efficient Video Distribution in Information-Centric Networks," IEEE Transactions on Multimedia, vol. 19, no. 6, 2017, pp. 1299-1312. DOI: 10.1109/TMM.2017.2652738.

[27] Leila Abadi, Amira Khalid, Predictive Maintenance in Renewable Energy Systems using Machine Learning , Machine Learning Applications Conference Proceedings, Vol 3 2023.

[28] X. Zhang, et al., "A Survey on Distributed Hash Table-based Storage Systems," Journal of Network and Computer Applications, vol. 40, 2014, pp. 312-340. DOI: 10.1016/j.jnca.2013.12.001

[29] Y. Zeng, P. Zhang, Y. Xiao, and X. Zhang, "A Survey on Fairness in Resource Allocation for Wireless Networks: Methods, Measurements, and Challenges," IEEE Communications Surveys & Tutorials, vol. 18, no. 2, 2016, pp. 1323-1350. DOI: 10.1109/COMST.2015.2502918