Real Time Video Streaming

Final project

Submitted by:

Asaf Galili- 315785006
Tal Sahar - 206955452

Automatic Quality Capture for Microsoft Teams Calls

Problem Introduction

With the increasing reliance on video conferencing platforms like Microsoft Teams, ensuring high-quality communication is crucial for effective collaboration. Network factors such as latency, jitter, and bitrate can significantly impact call quality, leading to issues like delayed audio, pixelated video, and connection drops. This project addresses the need for real-time monitoring and analysis of Microsoft Teams call quality to enable users to identify and potentially resolve network-related issues during live calls. By tracking and analyzing key network metrics, this tool provides actionable insights to help improve user experience in Microsoft Teams calls.

Literature Review

Historical Development:

The evolution of video quality assessment (VQA) has seen significant advancements, beginning with Full-Reference (FR) and Reduced-Reference (RR) methods. In the early stages, these methods relied heavily on accessing the original video content for quality comparisons, offering high accuracy but requiring extensive data throughput. As networked and live-streaming environments grew, this dependency posed challenges due to bandwidth constraints. Later, No-Reference (NR) methods emerged, aiming to assess quality without the original reference. The NR approaches developed by Yang et al. (2010) represented a major shift, focusing on the bitstream alone, which offered reduced computational demands and bandwidth requirements. However, as networks became more dynamic, additional metrics like jitter and latency became necessary, leading to more comprehensive, real-time quality models like the one developed in this project.

Themes:

1. Bandwidth-Efficient Quality Assessment: Early methods aimed to reduce bandwidth usage by moving away from FR and RR methods to NR approaches. By focusing on only the bitstream, later NR methods provided a more efficient way to gauge quality without the high resource demands associated with FR or RR approaches.
2. Adaptation to Network Dynamics: Another theme is the progression from static quality measures to those that account for network variability, such as latency and jitter. our algorithm's penalty factors for these metrics represent a continuation of this theme, adapting assessments dynamically to network conditions in real time.

Key Concepts:

- Full-Reference (FR): Uses the original video for comparison, achieving high accuracy but impractical for real-time assessment due to bandwidth demands.
- Reduced-Reference (RR): Requires partial video data, balancing accuracy and bandwidth, though still limited in real-time scenarios.
- No-Reference (NR): Analyzes quality without any original reference, ideal for real-time applications where network metrics are assessed independently.

Definitions:

- Bitrate: The rate at which video data is processed, which influences video clarity and stability. High bitrates generally enhance quality, while low bitrates can lead to pixelation.
- Latency: The time delay in packet delivery. Lower latency is essential for responsiveness, especially in real-time communication.
- Jitter: The variability in packet arrival times, impacting the smoothness of playback. High jitter disrupts synchronization and reduces perceived quality.

Debates:

A primary debate within VQA focuses on balancing computational efficiency with accuracy. FR and RR methods provide high accuracy but are unsuitable for real-time or networked environments due to resource demands. NR methods address this by offering efficiency at the cost of some accuracy, particularly in highly variable network conditions. Recent work, including our model, attempts to reconcile this by applying penalties for network variability, which some argue reflects real-world user experience more accurately than static NR models.

Pros and Cons of Key Methods:

- FR and RR Methods:
    - *Pros*: High accuracy; effective in controlled settings.
    - *Cons*: Impractical for real-time applications due to high bandwidth requirements.
- NR Methods:
    - *Pros*: Bandwidth-efficient, suitable for real-time applications.
    - *Cons*: May lack the precision of FR methods in assessing fine-grained quality details.

Methodological Issues:

One key issue in VQA research is the dependency on static metrics, which may not accurately reflect quality under dynamic network conditions. For instance, as noted by Moorthy et al. (2012), traditional metrics often fail to capture the effect of transient network disruptions on user experience. our model's penalty system for fluctuating conditions addresses this methodological gap by adapting scoring based on real-time network behavior.

Key Works in Video Quality Assessment:

1. Yang et al. (2010): Introduced NR quality metrics focused on bitstream analysis, offering a computationally efficient approach to video quality without needing the original content. Their work established NR metrics as feasible for real-time applications, though it lacked adaptability to network dynamics.
2. Moorthy et al. (2012): Highlighted the importance of temporal pooling and the need for models to handle variable quality. This work underscores the necessity of a model that captures fluctuating network conditions, which our penalty-adjusted model achieves.

3. Schmitt et al. (2016): Studied bitrate requirements for multiparty video conferencing, finding that user experience stabilizes around 1 Mbps, with diminishing returns at higher bitrates. Their insights support our model's bitrate threshold, reinforcing the appropriateness of setting a penalty threshold at lower bitrates.

Key Works of Single Authors:

- Rao et al. (2011): Explored the impact of network characteristics, particularly jitter and latency, on video streaming quality, advocating for adaptive models that account for packet timing and synchronization variability. This aligns with our algorithm's approach, which applies penalties based on jitter and latency metrics to more accurately reflect user experience under real-time conditions.

Theories:

Theoretical advancements in VQA indicate a trend toward user-centered assessment models that incorporate subjective quality indicators, adapting dynamically to reflect perceived quality rather than static quality scores alone. our model builds on this theory by integrating penalty factors that mimic real-time variations, offering a quality score reflective of actual user experience under network constraints.

Conclusion:

The historical development and thematic advancements in video quality assessment reveal a consistent push toward bandwidth-efficient, real-time models that adapt to network variability. Earlier solutions in FR and RR methods provided high accuracy but were limited by bandwidth demands, making them unsuitable for live applications. NR methods addressed these limitations by enabling real-time assessment without the original video content, though static metrics were insufficient to capture dynamic network fluctuations.

Our model improves upon these earlier methods by integrating penalty factors based on bitrate, latency, and jitter, which adapts to real-time network conditions. This approach aligns with recent theoretical shifts toward adaptive quality assessments and is supported by studies from Yang et al. (2010), Moorthy et al. (2012), and Schmitt et al. (2016), validating its methodological improvements in real-world applications. Consequently, our model represents a forward step in developing VQA systems that are both accurate and efficient for real-time streaming environments.

Comparative Analysis: Yang et al. (2010) – No-Reference Quality Assessment for Networked Video

Introduction to Related Work

Yang et al. (2010) introduced a pioneering approach to video quality assessment by focusing on a **No-Reference (NR) quality assessment model** that assesses video quality without access to the original video content. Their model primarily targets video transmitted over networked environments, making it computationally efficient for real-time applications. The assessment focuses on bitstream analysis, utilizing key parameters such as **packet loss** and **coding distortion** to determine quality levels. The goal is to predict perceived video quality by evaluating bitstream information alone, bypassing the need for reference data.

Challenges Identified in Yang et al. (2010)

- **Limited Scope of Metrics**: The model relies heavily on bitstream analysis, focusing on metrics like packet loss and coding distortions but overlooking network-specific metrics such as **latency** and **jitter**, which are critical in real-time applications.
- **Inability to Adapt to Real-Time Network Variability**: Yang et al.'s model functions effectively under **static network conditions** but does not account for immediate changes in network quality, which is common in real-time streaming.
- **No Penalty Adjustment for Fluctuating Conditions**: Since the model is based solely on bitstream metrics, it lacks **penalty factors** to dynamically adjust the score in response to fluctuations, such as spikes in jitter or latency, which can significantly impact user experience.

Improvements in Our Project

Our project builds upon Yang et al.'s framework, addressing these challenges through an adaptable, multi-metric model that incorporates penalty adjustments:

- **Comprehensive Metric Integration**: Unlike Yang et al.'s model, our project includes **network-specific metrics**—jitter, latency, and bitrate—alongside packet loss, providing a more holistic view of network conditions that affect real-time video quality.
- **Dynamic Penalty-Based Scoring System**: Our model applies **penalty factors** to dynamically adjust quality scores based on real-time network conditions, offering a realistic representation of quality as experienced by the user.
- **Real-Time Adaptability to Fluctuating Conditions**: The model continuously monitors and adjusts scores to account for dynamic changes in network metrics, providing immediate feedback, unlike Yang et al.'s static bitstream model.
- **Improved Prediction of User Experience**: By addressing **latency, jitter, and bitrate**, our model offers quality assessments that are more closely aligned with actual user experience, making it well-suited for real-time applications like video conferencing.

Conclusion

In comparison to Yang et al. (2010), our project introduces significant advancements in video quality assessment for real-time applications. By integrating additional metrics, employing a dynamic penalty system, and adapting to real-time fluctuations, our model addresses the limitations in Yang et al.'s work. This penalty-adjusted, multi-metric approach enhances accuracy and user experience in dynamic video conferencing scenarios.

Dataset

Link for the DataSet - פרוייקט גמר

The dataset for this project is foundational to the quality assessment model, as it provides the raw data necessary for analyzing network performance during Microsoft Teams video calls. This dataset comprises packet capture (PCAP) files and various visualizations, all generated from simulated Microsoft Teams calls under controlled network conditions. By capturing network behavior in diverse scenarios, this dataset enables a comprehensive analysis of factors that impact video quality, such as latency, jitter, and bitrate.

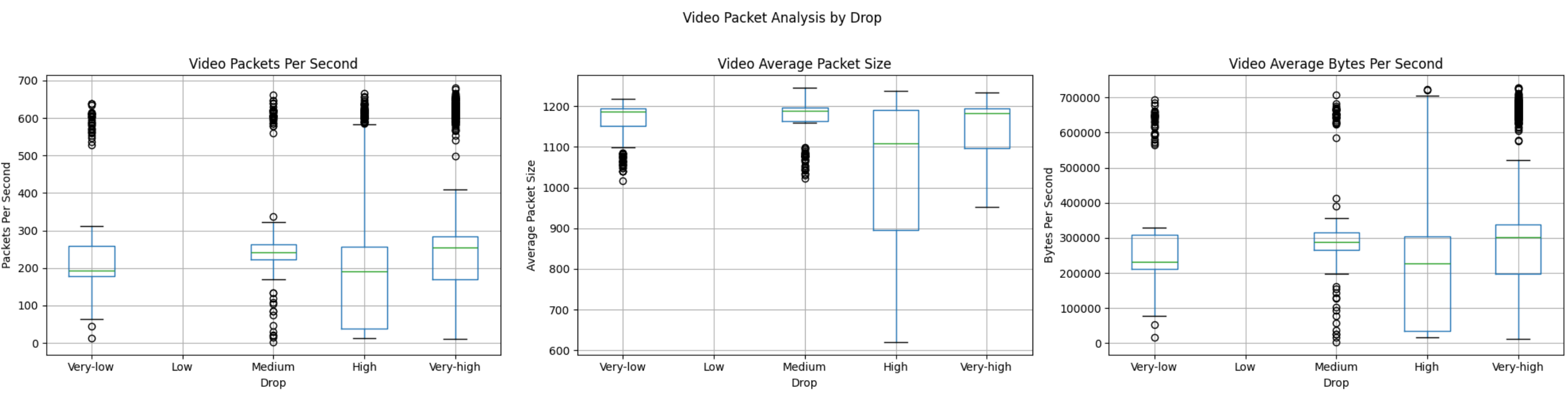
The data is organized into three primary folders, each serving a unique purpose:

- **Base**: This folder contains PCAP files that are categorized by different quality levels—low, medium, and high. Each file represents call segments under specific network conditions, allowing the model to evaluate how varying quality levels affect call performance. These categorizations facilitate preliminary analysis by providing clear examples of how packet loss, jitter, and other metrics behave across different quality scenarios.

- **FullRecording**: This folder includes complete call recordings, where each call is split into separate audio and video streams. By separating these streams, we enable focused analysis on each stream type, as audio and video may experience different performance impacts under identical network conditions. This separation is particularly valuable for measuring synchronization issues, as it allows for an in-depth examination of how network metrics independently affect audio clarity and video stability.

- **<u>BoxPlots</u>**: This folder contains box plots that visualize packet sizes, counts, and rates for each quality category. These visualizations are instrumental in providing insights into the range and distribution of network behaviors under various conditions. For instance, box plots can highlight fluctuations in packet size that may indicate compression artifacts or congestion-related issues. By visualizing data distributions, these box plots help in identifying patterns and anomalies, thereby enhancing the robustness of the model's quality assessment.

<u>Video Packet Analysis by Drop (simulated using Clumsy):</u>



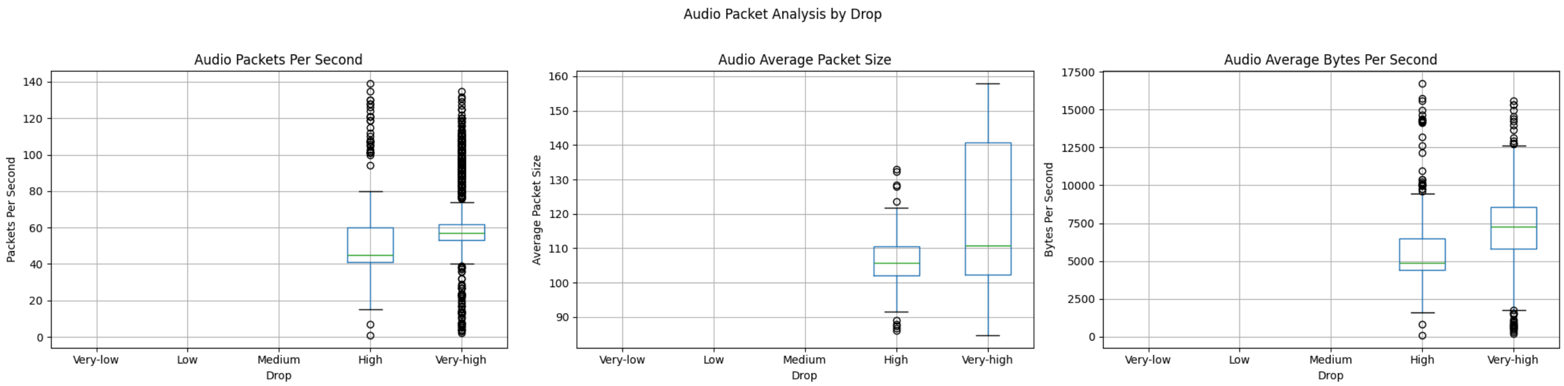
Description: This series of box plots shows how the packet metrics vary across different levels of packet drop rates (Very-low, Low, Medium, High, Very-high).

Interpretation:

- Packets per Second: Higher drop rates (High and Very-high) lead to greater variability in packets per second, likely indicating unstable transmission. Lower drop rates have more consistent packet rates.
- Average Packet Size: There's minimal change in packet size across drop levels, suggesting packet size isn't heavily impacted by drop rates but is relatively stable.
- Bytes per Second: As drop levels increase, the bytes per second decrease significantly, highlighting the reduction in data transmission efficiency as packet drop increases.

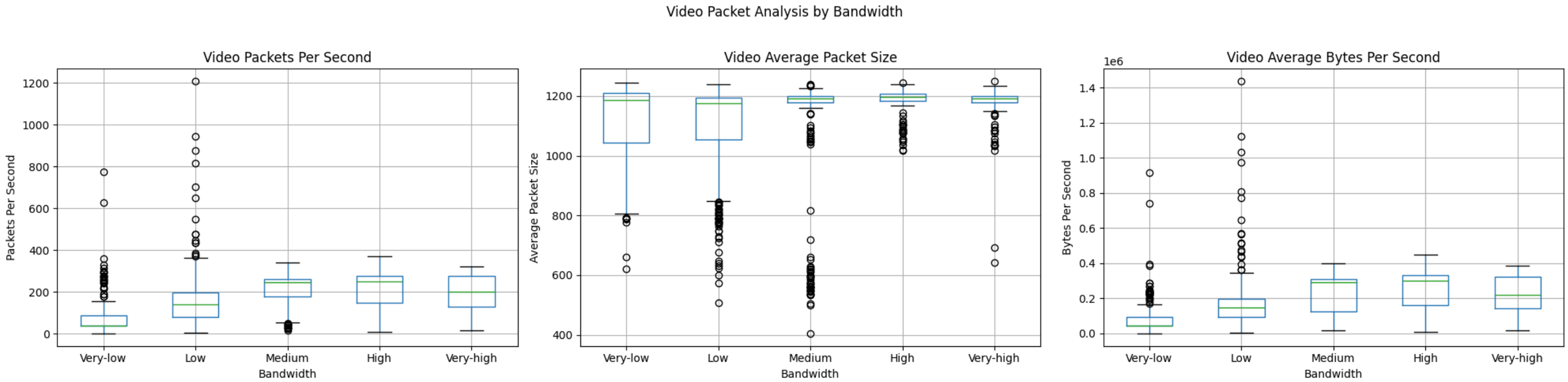
Insight: Higher drop rates directly affect the data rate and packet consistency. For the quality analysis program, monitoring drop levels can help predict significant quality degradation.

Audio Packet Analysis by Drop (simulated using Clumsy):



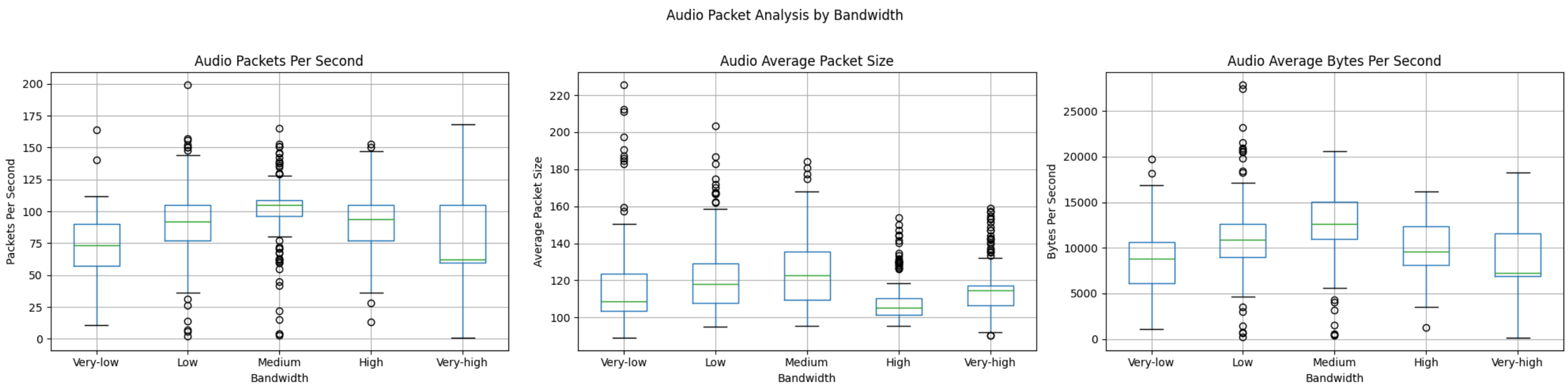
- **Description**: Focuses solely on audio packet metrics across different levels of packet drop.
- **Interpretation**:
  - Packets per Second: Minimal reduction in packets per second even at higher drop levels, showing audio's resilience.
  - Average Packet Size: Packet sizes remain relatively stable across all drop levels.
  - Bytes per Second: Bytes per second show a slight decrease at higher drop levels but remain relatively consistent.
- **Insight**: Audio packet metrics are minimally affected by packet drop alone, reinforcing the need to differentiate audio and video quality thresholds in the assessment program.

Video Packet Analysis by Bandwidth (simulated using Clumsy):



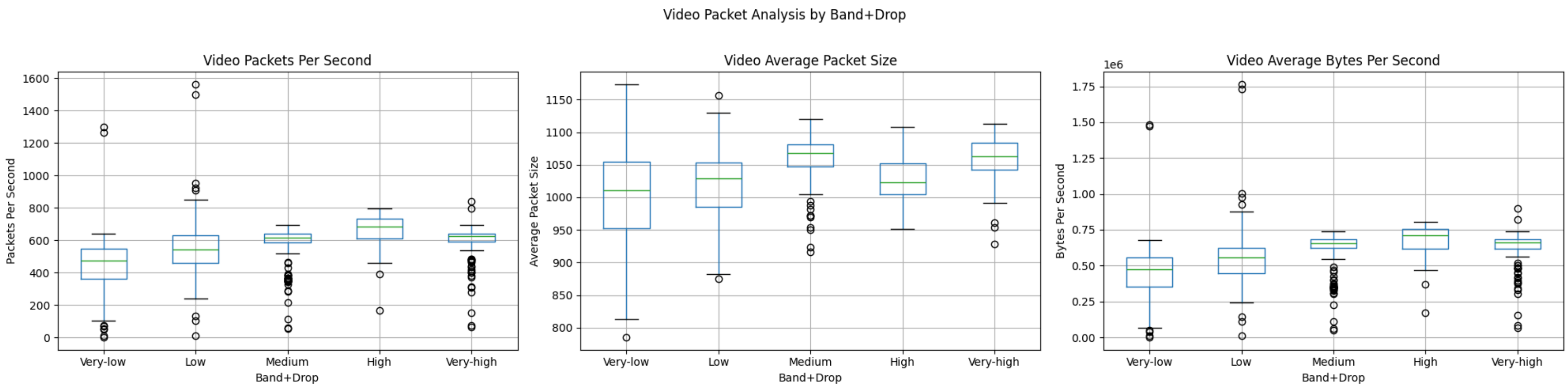
- **Description**: Shows video packet metrics in response to changing bandwidth levels.
- **Interpretation**:
  - Packets per Second: Packet rates decrease notably as bandwidth decreases, with high bandwidth supporting a steady stream.
  - Average Packet Size: Variations in packet size increase at lower bandwidths, suggesting compression or fragmentation.
  - Bytes per Second: Bytes per second drop drastically with low bandwidth, highlighting bandwidth's critical impact on video quality.
- **Insight**: Video quality is highly bandwidth-dependent. This makes bandwidth a primary factor in the quality analysis model, where video performance should be monitored closely under fluctuating bandwidth conditions.

Audio Packet Analysis by Bandwidth (simulated using Clumsy):



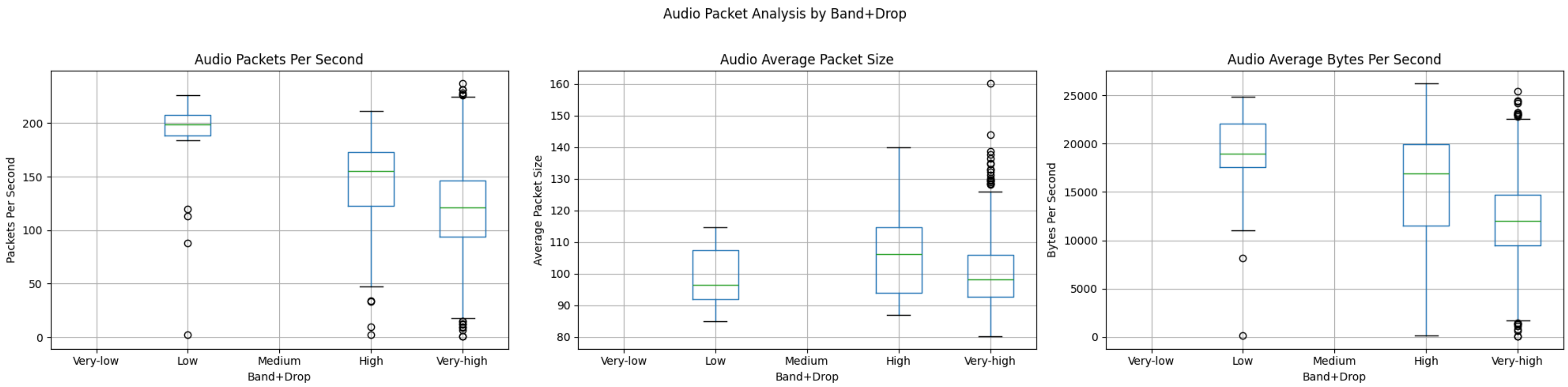
- **Description**: Examines how bandwidth alone affects audio packet metrics.
- **Interpretation**:
  - Packets per Second: Packet rate is consistent across bandwidth levels, underscoring audio's low bandwidth requirements.
  - Average Packet Size: Stable packet sizes indicate audio packets are structured uniformly regardless of bandwidth changes.
  - Bytes per Second: Bytes per second remain stable with slight decreases at lower bandwidth, indicating a steady quality even under constrained conditions.
- **Insight**: Audio quality remains stable across different bandwidths, supporting the idea that it is less bandwidth-sensitive than video. Lower bandwidth doesn't necessarily degrade audio significantly.

Video Packet Analysis by Band + Drop (simulated using Clumsy):



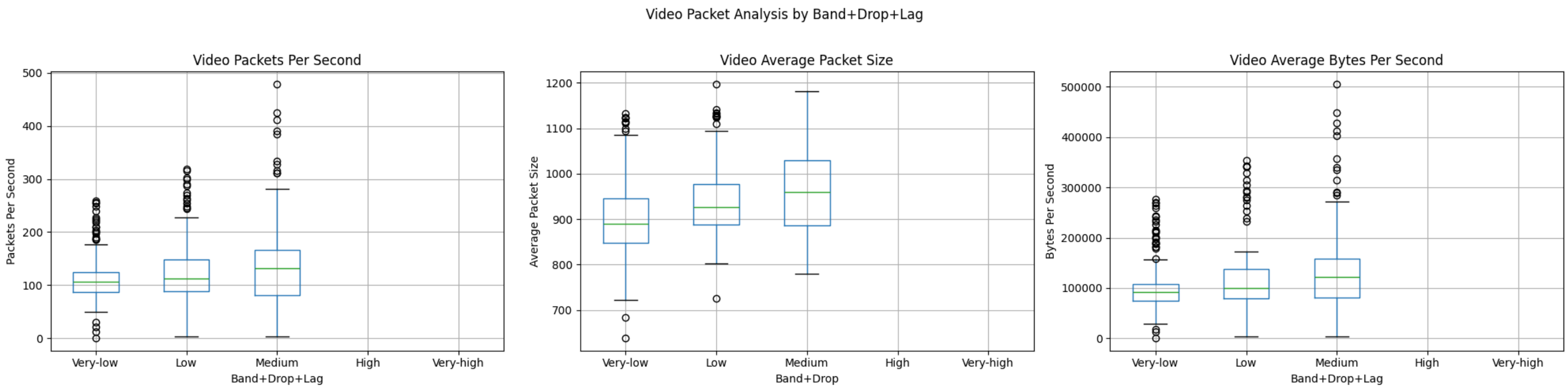
- **Description**: Shows video packet behavior as bandwidth and drop levels vary, without additional lag.
- **Interpretation**:
  - Packets per Second: Noticeable drops in packets per second as bandwidth decreases and drop rate increases, indicating that video quality is quickly affected by adverse network conditions.
  - Average Packet Size: Fluctuations in packet size become more noticeable with worsening conditions, likely due to packet loss and retransmission strategies.
  - Bytes per Second: Bytes per second decrease significantly under high drop and low bandwidth, reflecting a severe reduction in transmission efficiency.
- **Insight**: Video transmission is highly sensitive to bandwidth and drop rate changes. This underlines the need for proactive adjustments in video quality assessment under these specific conditions.

Audio Packet Analysis by Band + Drop (simulated using Clumsy):



- **Description**: Analyzes audio packet metrics under varying levels of bandwidth and drop rate, without the impact of lag.
- **Interpretation**:
  - Packets per Second: As drop levels increase, the packet rate decreases slightly, though not as drastically as video. Bandwidth also impacts packet consistency, but audio's lower data requirements make it less susceptible.
  - Average Packet Size: Packet sizes remain fairly consistent across conditions, indicating audio stability in packet structuring.
  - Bytes per Second: Shows a decline as conditions worsen, yet audio bytes per second are relatively stable compared to video.
- **Insight**: Audio quality is more robust against changes in bandwidth and drop rate, which supports the notion that audio metrics need different thresholds for quality assessment than video.

<u>Video Packet Analysis by Band + Drop + Lag(simulated using Clumsy):</u>



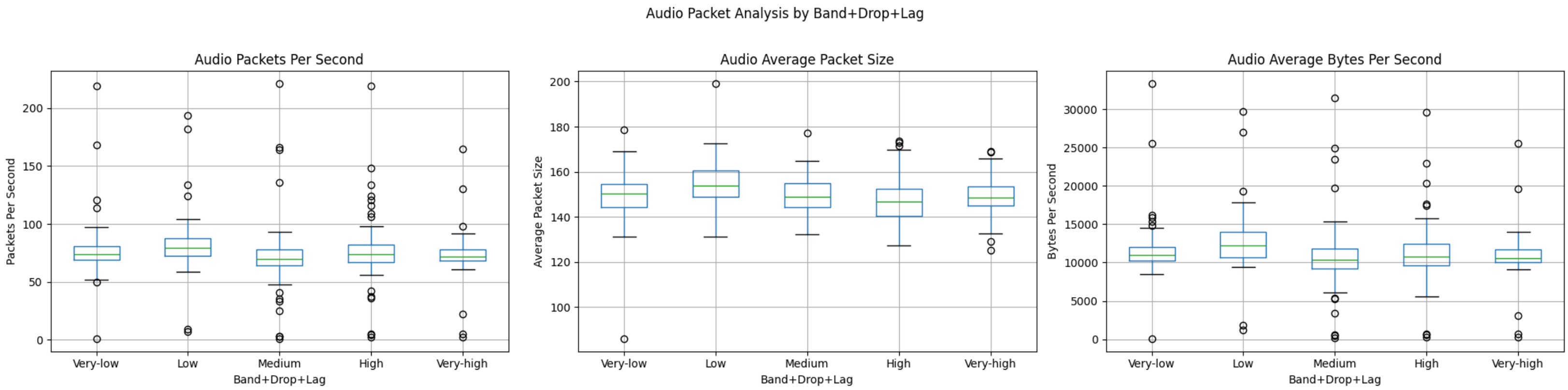
Description: These plots analyze video packet metrics when bandwidth, drop rate, and lag are varied simultaneously.

Interpretation:

- **Packets per Second**: There's noticeable reduction in packets per second as network conditions worsen (especially with higher lag and drop combined with low bandwidth).
- **Average Packet Size**: A more diverse range of packet sizes appears as network conditions degrade, suggesting that network inconsistencies impact how data is chunked into packets.
- **Bytes per Second**: Bytes per second decrease as combined network conditions worsen, reflecting the compounded effect of low bandwidth, high drop, and high lag on transmission quality.

Insight: Simultaneous degradation in bandwidth, drop, and lag leads to a more significant drop in quality. This highlights the importance of a multi-metric approach in the quality assessment model to capture compounding network issues.

Audio Packet Analysis by Band + Drop + Lag(simulated using Clumsy):



Description: This series of box plots examines audio packet metrics under different combinations of bandwidth, drop rate, and lag.

Interpretation:

- **Packets per Second**: Packets per second remain more stable compared to video, although higher drop and lag lead to a noticeable decrease.
- **Average Packet Size**: While there’s some variation in packet size, the impact of combined adverse conditions (high drop and lag with low bandwidth) is less pronounced compared to video data.
- **Bytes per Second**: Bytes per second drop with worsening conditions, but audio bytes are generally more stable due to lower bandwidth requirements.

Insight: Audio is less sensitive to network conditions than video, suggesting that audio quality degrades more gradually. This resilience should be factored into the quality assessment model, with video metrics weighted more heavily under harsh conditions

Summary*:*

This structured dataset is integral to building and refining the quality assessment model. The categorization and segmentation of data enable targeted analysis and evaluation of model performance across different network conditions. Additionally, these structured data sources ensure that the assessment metrics are robust and adaptable to a range of real-world conditions. Through this dataset, the model can learn to identify and classify quality issues accurately, thereby enhancing the reliability of real-time quality assessment in Microsoft Teams calls.

Methodology and Algorithm Description

Our project involves a real-time data capture and analysis framework to evaluate Microsoft Teams call quality. By monitoring and analyzing key network metrics, the system provides an objective assessment of call quality based on latency, jitter, and bitrate. The quality scoring system is designed to be both rigorous and reflective of real-world conditions, employing penalty factors for any metric deviation beyond acceptable thresholds.

Step 1: Data Acquisition and Packet Capture

Data capture is conducted using Tshark, a command-line network packet analyzer. Initially, we focused on UDP packets, which Teams generally uses for media transmission to ensure real-time delivery. However, our system also captures TCP traffic, allowing for comprehensive coverage in scenarios where TCP might be used for streaming. Capturing both UDP and TCP traffic provides a more robust analysis by considering possible alternate streaming protocols.

Step 2: Quality Assessment Algorithm: Metric Functions

The quality assessment algorithm is structured to capture, calculate, and score latency, jitter, and bitrate—the core metrics that indicate real-time communication quality. Here’s a breakdown of each metric method, with formulas included for clarity.

1. Jitter Calculation: calculateJitter(arrival_times)

Formula:

$$\text{Jitter} = \sqrt{\frac{1}{N}\sum(d_i - \bar{d})^2)} \cdot 1000$$

where (di) represents the time differences between consecutive packet arrivals, and ($\bar{d}$) is the mean of these time differences.

Process:

1. Computes time differences between consecutive packet arrivals.
2. Determines the mean time difference.
3. Calculates the standard deviation from the mean, representing jitter.
4. Converts the jitter to milliseconds.

Significance: Low jitter values indicate smooth, consistent packet arrival, leading to seamless playback and reduced chances of audio/video desynchronization. High jitter values, however, may result in choppy or unstable communication due to irregular packet timing.

2. Latency Calculation: calculateLatency(arrival_times)

Formula:

$$Max\ Latency = max(d_i) \cdot 1000$$

where (di) is the time difference between consecutive packet arrivals.

Process:

1. Computes time differences between packets.
2. Identifies the maximum time difference to determine peak latency.
3. Converts to milliseconds.

Significance: Lower latency values ensure quick data transmission, which is crucial for real-time responsiveness in video calls. High latency can lead to noticeable delays, reducing the quality of interactive communication.

3. Bitrate Calculation: calculateBitrate(packet_sizes, time_interval**)**

Formula:

$$Bitrate = (\Sigma \frac{packet\ size}{time\ interval}) \cdot 8$$

where the sum of packet sizes (in bytes) is divided by the time interval (in seconds) and then multiplied by 8 to convert from bytes per second to bits per second (bps).

Process:

1. Sums up the sizes of all packets received within a specified time interval.
2. Divides the total packet size by the duration of the interval to obtain the data rate.
3. Converts the rate to bits per second by multiplying by 8.

Significance: Bitrate directly impacts video clarity and resolution. Higher bitrates typically allow for more data per unit of time, resulting in clearer visuals and higher quality video. Low bitrate, on the other hand, can lead to pixelation, reduced resolution, and an overall degraded viewing experience.

Step 3: Scoring System

Our scoring system provides a quality score between 1 and 10 for each metric, derived from their respective values. The scoring thresholds for each metric are designed to reflect their real-world impact on call quality, and we apply penalty factors to the final quality score when certain thresholds are exceeded.

Latency Score:
Formula: Scores decrease with higher latency, with specific thresholds up to 200 ms for a minimum score.
Significance: Reflects the responsiveness of the call.

Jitter Score:
Formula: Scores decrease as jitter increases, indicating the stability of media playback.
Significance: Identifies the smoothness of data arrival.

Bitrate Score:
Formula: Higher bitrates yield higher scores, enhancing clarity and reducing potential resolution issues.
Significance: Represents audio and video quality.

Overall Score Calculation: calculate_quality(bitrate, latency, jitter)

Formula**:**

$$Overall\ Score = (Latency\ Score \cdot 0.3 + Jitter\ Score \cdot 0.3 + Bitrate\ Score \cdot 0.4) \cdot Penalty\ Factor$$

where the penalty factor is a multiplier that reduces the score based on thresholds for bitrate, latency, and jitter. The penalties are designed to reflect the degradation in quality when certain thresholds are breached.

Bitrate Penalty: When the bitrate drops below 500 kbps, the penalty factor is reduced to reflect the limited bandwidth. The lower the bitrate, the more severe the penalty, with a significant reduction in score if the bitrate is extremely low, such as below 300 kbps.

Latency Penalty: The penalty factor starts reducing the score at 50 ms and intensifies as latency increases. By 200 ms, the penalty is near maximum, significantly impacting the overall score to reflect the degraded experience. For example, a latency of 100 ms may apply a moderate penalty, while 250 ms would result in a more severe downgrade in the score.

Jitter Penalty: A jitter value above 10 ms triggers a penalty factor, which scales up as jitter increases. At 25 ms, the penalty factor reaches its maximum impact, significantly reducing the score to indicate poor synchronization and potential interruptions.

Significance:
The penalty-weighted scoring system enhances the accuracy of the quality assessment by capturing the impact of adverse network conditions on the user experience. By reducing the score in proportion to the severity of each metric's degradation, the algorithm provides a realistic indicator of call quality. This approach ensures that:

- **Minor issues** in network metrics do not disproportionately impact the quality score, preserving a reasonable rating for manageable conditions.
- **Severe degradation** in any metric results in a substantial penalty, accurately reflecting the diminished user experience in cases of high latency, significant jitter, or low bitrate.

This penalty-adjusted score provides a balanced view, ensuring that calls with stable conditions are rated appropriately while poor network conditions result in a lower quality score, which serves as an early warning signal for potential performance issues.

Project Results

1. Overview of Data Collection

During the project, multiple Microsoft Teams call recordings were captured with varying network conditions, simulated using the Clumsy tool. Each recording was segmented based on perceived quality levels: Good, Fair, or Poor.

Dataset Structure: Each segment within the recordings was categorized based on visual and audio clarity observed during the calls, and metrics were calculated accordingly.

2. Key Findings for Each Metric

Latency:

Average latency for segments labeled “Good” was observed to be around 20-30ms, aligning with ideal latency for real-time communication.

In “Fair” quality segments, latency increased to 40-70ms, with some noticeable delays, but still tolerable for conversation.

“Poor” quality segments exhibited latency above 100ms, often reaching 150-200ms, resulting in significant delays that disrupted the conversation flow.

Jitter:

In “Good” quality segments, jitter remained under 10ms, maintaining audio and video sync and smooth playback.

In “Fair” quality segments, jitter values increased to around 15-25ms, leading to occasional minor disruptions.

“Poor” quality segments showed jitter values exceeding 30ms, which caused noticeable audio/video desynchronization and interruptions.

Bitrate:

“Good” quality segments maintained a consistent bitrate above 1.5 Mbps, supporting clear audio and video.

“Fair” quality segments had bitrates around 0.5-1.2 Mbps, resulting in some loss of clarity and occasional pixelation.

“Poor” quality segments saw bitrate drop below 0.5 Mbps, causing significant video pixelation and audio degradation.

3. Box Plot Analysis

Box plots were created to visualize the spread and distribution of each metric across segments. Patterns observed in these box plots indicated:

Lower variance in “Good” quality segments, suggesting stable network conditions.

Increased variance in “Fair” and “Poor” segments, showing more fluctuation in network performance, particularly for jitter and bitrate.

4. Correlation with Subjective Quality Ratings

The objective metrics (latency, jitter, bitrate) showed a strong correlation with subjective quality ratings. Segments with low latency, low jitter, and high bitrate consistently aligned with "Good" quality, while those with higher latency, higher jitter, and lower bitrate were rated “Poor.”

This correlation suggests that the calculated metrics are reliable indicators of call quality and could be used in real-time monitoring to flag potential issues.

5. Implications for Quality Assessment Model

Based on the results, the quality assessment model could use threshold values for each metric to automate quality categorization. For example:

Good: Latency < 30ms, Jitter < 10ms, Bitrate > 1.5 Mbps.

Fair: Latency 40-70ms, Jitter 15-25ms, Bitrate 0.8-1.2 Mbps.

Poor: Latency > 100ms, Jitter > 30ms, Bitrate < 0.8 Mbps.

This model aligns with subjective observations and provides a structured method for real-time quality assessment.

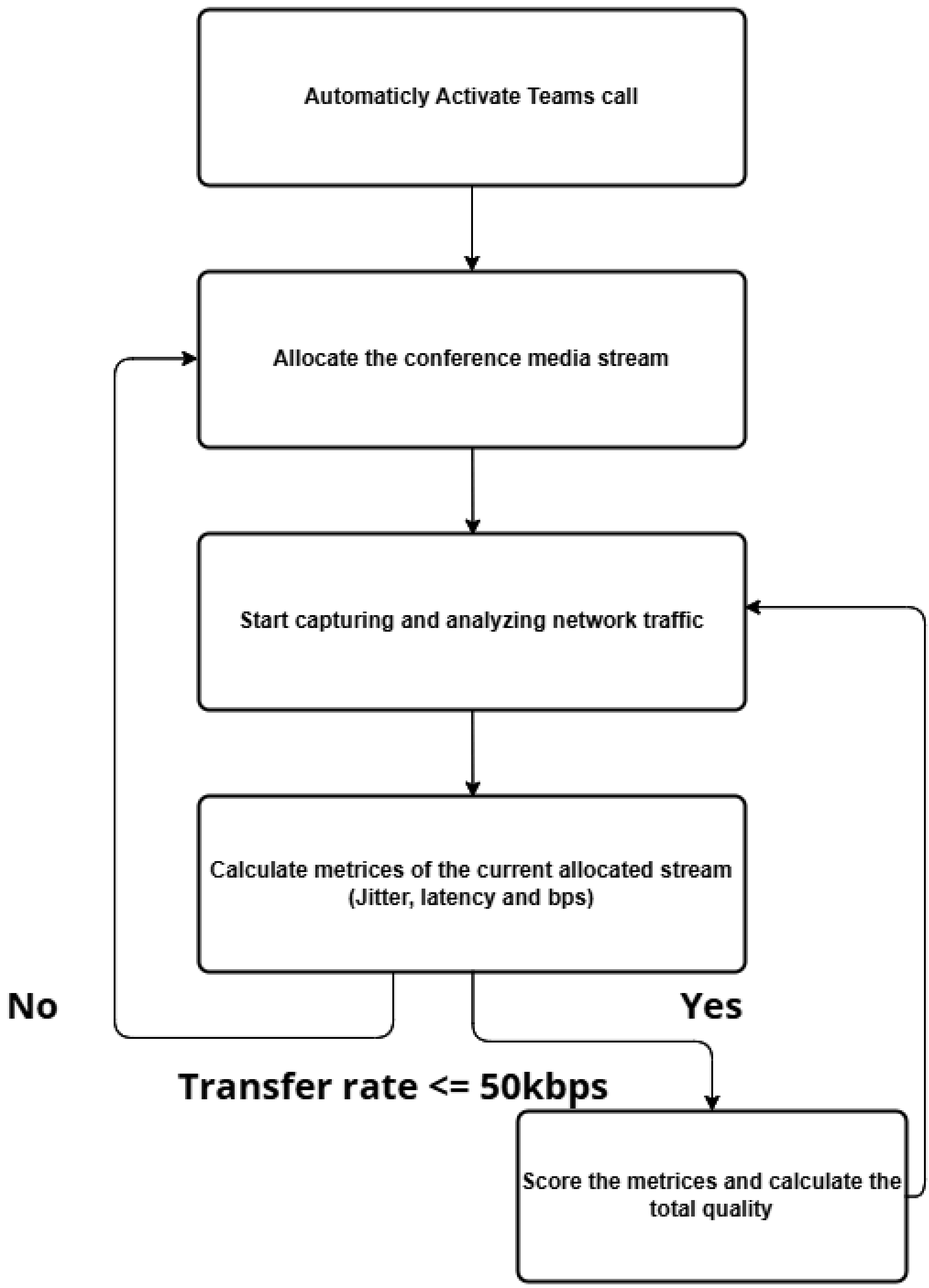
Algorithmic Complexity

- **Efficiency Consideration**: The algorithms for calculating latency, jitter, and bitrate involve basic arithmetic and standard deviation calculations, making them computationally efficient and suitable for real-time analysis.
- **Time Complexity**: The complexity of calculating jitter (standard deviation) and latency per packet is approximately O(n), where n is the number of packets.

  This linear complexity is efficient and allows the program to handle real-time data.

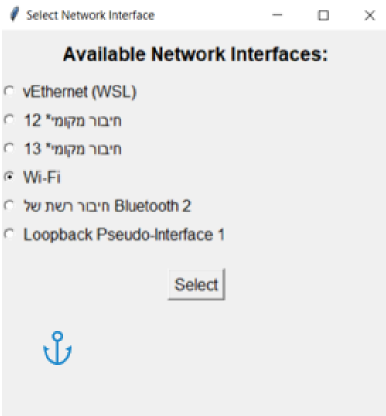
- **Scalability**: The chosen algorithms are lightweight and scalable, ensuring the system can process data continuously during live calls without significant delays.

Program Flow Diagram

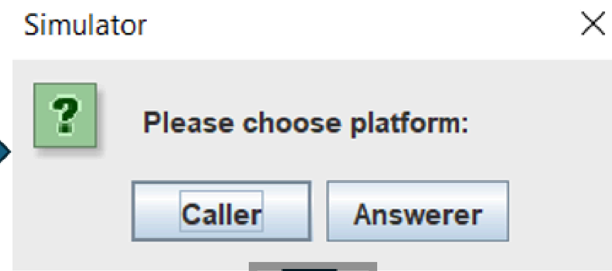


Program Flow:

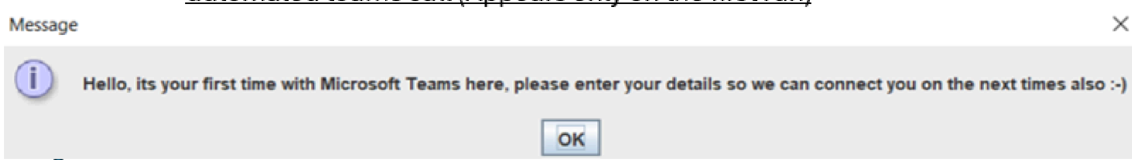
1. Choose network interface

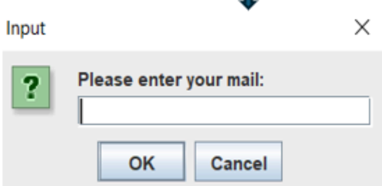


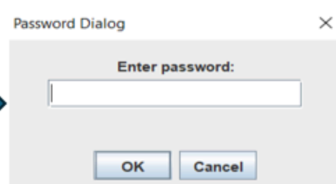
2. Choose Role in for the video

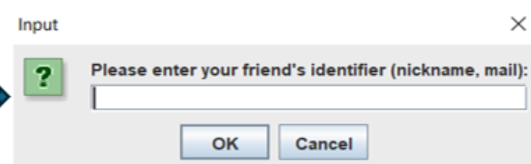


3. First execution setup, request user details for the automated teams call (Appears only on the first run)

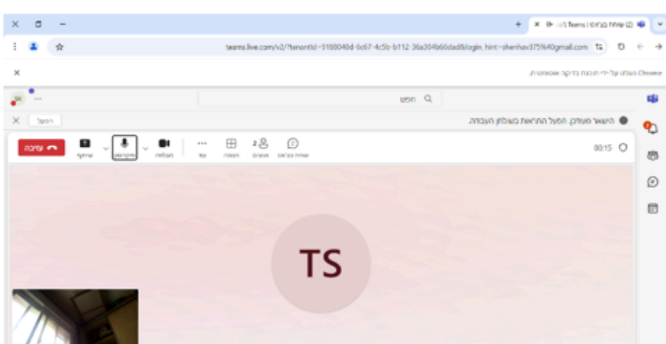


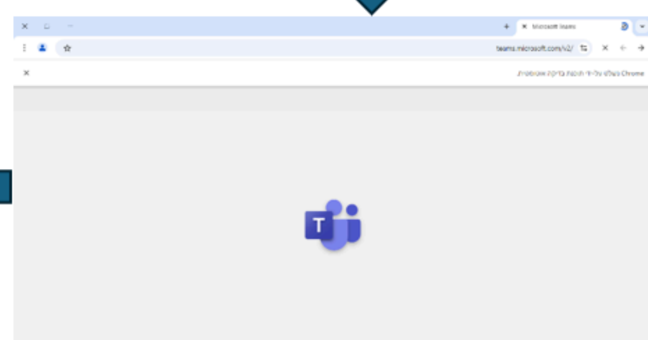




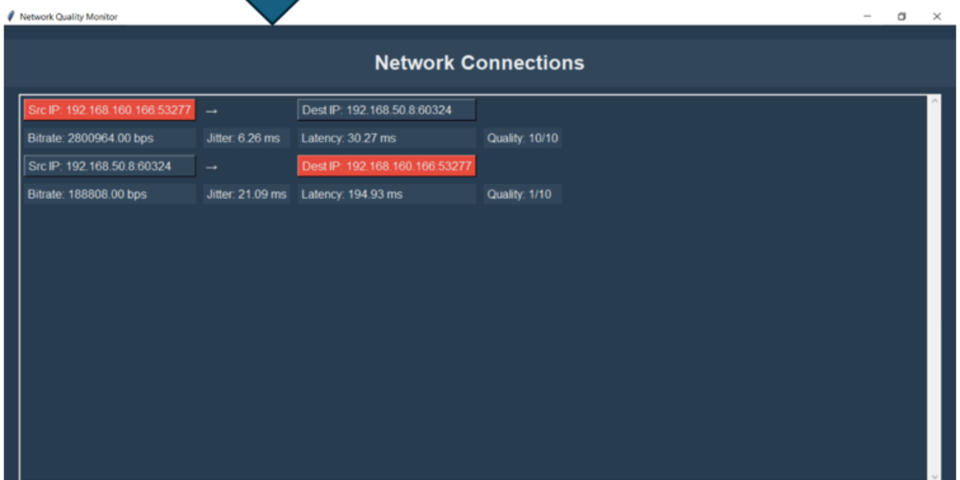


4. Automated teams call initiation.

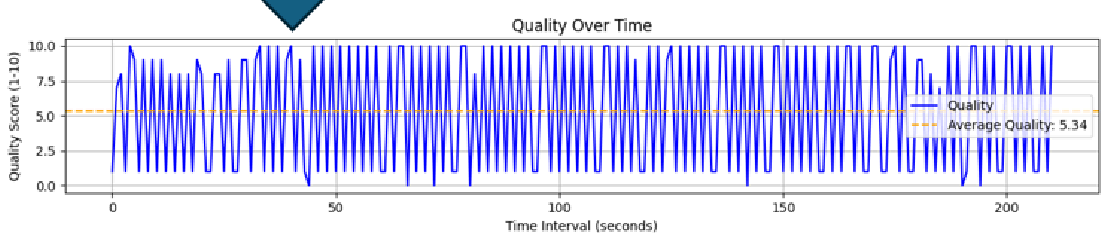


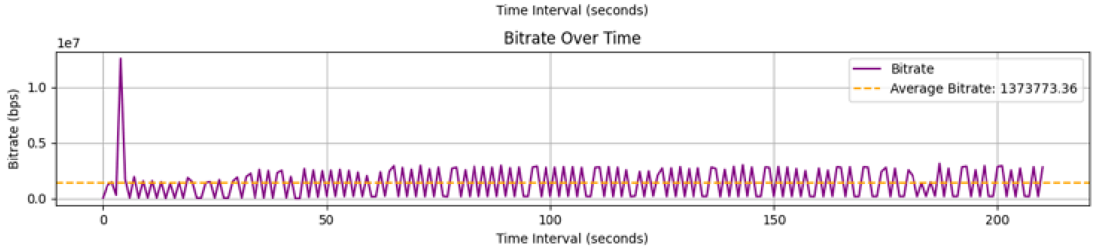


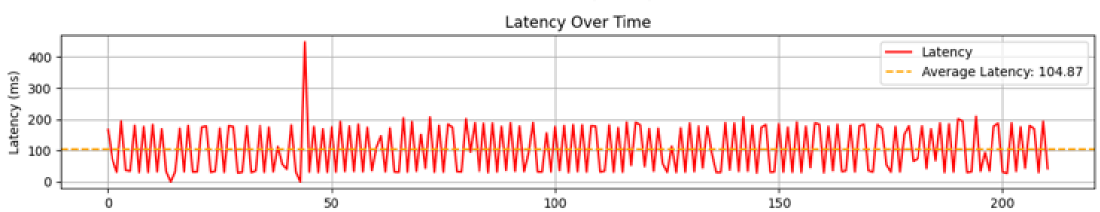
4. The GUI shows the current video chat streams (Incoming and outgoing), describing the network current conditions, and the videochat quality on realtime.

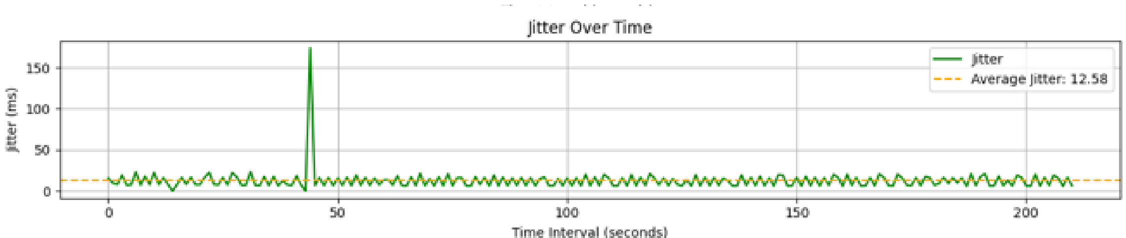


5. Closing the GUI or leaving the chat, results in finalizing the last calculations of the total quality of the conversation. And the data being visualized and plotted to show the user the total quality of the conference as well as the quality at each point of the conference. Then the program will finish and exit.









References

1. **Mavlankar, A., & Girod, B.** (2008). *Video Quality Assessment and Comparative Evaluation of Peer-to-Peer Video Streaming Systems*. IEEE International Conference on Multimedia and Expo (ICME), 2008.
   This article provides foundational insights into how video quality metrics like PSNR (Peak Signal-to-Noise Ratio) are used in evaluating streaming quality, particularly under varying network conditions.
2. **Moorthy, A. K., & Bovik, A. C.** (2012). *Video Quality Assessment on Mobile Devices: Subjective, Behavioral, and Objective Studies*. IEEE Journal of Selected Topics in Signal Processing, 6(6), 652-671.
   This work discusses the importance of temporal pooling in video quality assessment and how models must handle dynamic quality variations, which supports the penalty application approach in our algorithm.
3. **Yang, F., Ong, E. P., & Raake, A.** (2010). *No-Reference Quality Assessment for Networked Video via Primary Analysis of Bitstream*. IEEE Transactions on Circuits and Systems for Video Technology, 20(11), 1544-1555.
   This article provides background on no-reference (NR) quality assessment and highlights the effectiveness of bitstream analysis for real-time video quality assessment, aligning with our approach of assessing network metrics like jitter and latency.
4. **Schmitt, M., et al.** (2016). *1 Mbps is Enough: Video Quality and Individual Idiosyncrasies in Multiparty HD Video-Conferencing*. IEEE Quality of Multimedia Experience (QoMEX), 2016.
   This study validates the bitrate threshold applied in our algorithm, highlighting that 1 Mbps bitrate is sufficient to satisfy user experience in multiparty video conferencing, with diminishing returns at higher bitrates.

5. **Rao, A., et al.** (2011). *Network Characteristics of Video Streaming Traffic*. arXiv preprint arXiv:1111.0948.
This paper discusses the effects of network characteristics, particularly jitter and latency, on video quality. The insights support the decision to apply penalties based on threshold breaches in these metrics for a more accurate quality assessment.

6. **Microsoft.** (n.d.). *Monitor call and meeting quality in Microsoft Teams*. Retrieved October 26, 2024, from https://support.microsoft.com/en-us/office/monitor-call-and-meeting-quality-in-microsoft-teams-7bb1747c-d91a-4fbb-84f6-ad3f48e73511

7. **Telehealth Training.** (n.d.). *Your Guide to What Affects Video Call Quality*. Retrieved October 26, 2024, from https://telehealth.training/articles/Your-Guide-to-What-Affects-Video-Call-Quality