Video Summarization for Soccer Match

Review Report - 8

Uchit N M - PES1UG22CS661

Arya Gupta - PES1UG22CS111

Faculty Mentors:

Prof. Prasad H B prasadhb@pes.edu

Dr. Gowri Srinivasa gsrinivasa@pes.edu

Prof. Preet Kanwal preetkanwal@pes.edu

Table of Content

Background....4
Section 1 - Exploring RAG....6
The NAIVE Rag Approach....7
Advanced Chunking Strategy for RAG....9
Agentic RAG....10
Section 2 - Dataset for Finetuning Open source LLM....12
Dataset Curation Strategy....12
Formatting the Dataset....13
Link to the Dataset....15
Section 3 - Analysing Audio Cue for Sports Event Detection:....16
Motivation....16
Audio Analysis Approaches....16
1: Naive Threshold-Based Segmentation....16
2: Ranked Intensity Scoring....17
3: MFCC-Enhanced Scoring....17
Experimental Results....18
Analysis and Challenges....21
Section 4 - Conclusion & Future Work....24

List of Figures

Fig 1 - The Naive RAG Architecture..... 7
Fig 2 - Agentic Approach Depiction..... 10
Fig 3 . Dataset Format (JSON Format)..... 13
Fig 4 . Background and Vocal Sound of Portugal Vs Spain 2018 match.....22
Fig 5. Audio waveform of Portugal Vs Spain 2018 match commentary..... 23

List of Tables

Table 1. Playlists of the Datasets.....15
Table 2. Naive Approach - All Detected Excited Segments..... 18
Table 3. Top Ranked Segments Approach..... 19
Table 4 . Top Ranked Segments Approach MFCC Based.....20
Table 5. Sport Wise Analysis.....21

Background

This report shows the progress made between **June 16th and July 12th, 2025** towards building a system that generates soccer match video highlights given the full-length match video, using open-source Large Language Models (LLMs), particularly **Llama4**.

It was noted that the limited context window (i.e., the maximum number of tokens that open-source LLMs can process at a time) prevented the models from effectively handling entire match transcripts (typically spanning 30000 tokens). As a result, they failed to capture most events and their context accurately. To address this constraint, we investigated Retrieval-Augmented Generation (RAG) approaches to enhance the quality of the outputs. (Section 1: Exploring RAG).

The RAG approach is intended to help capture context by retrieving the most relevant portions of the input transcript. With this goal in mind, we explored three strategies:

1. Naive RAG – A straightforward method that retrieves segments using simple keyword- or score-based matching.
2. Advanced Chunking for RAG – This technique divides the transcript into larger “parent” sections and smaller “child” chunks, enabling the model to retrieve both specific details and their broader context.
3. Agentic RAG – A more sophisticated approach in which the model engages in multi-step reasoning to actively select and organize content for summarization.

We observed that the RAG approach had limitations in coverage, despite retrieving relevant context, the model often failed to integrate and synthesize the information effectively. Upon further analysis, we found that the issue was largely due to cognitive overload in Llama 4 — the model struggled to manage and reason over the large, diverse chunks of retrieved content simultaneously.

As a result, we redirected our efforts toward fine-tuning the open-source LLM. As an initial step, we curated a custom dataset of 100 full-length soccer matches (see Section 2: Dataset Creation), comprising:

- 72 international matches
- 28 academy-level matches

The dataset preparation took approximately one week. Leveraging this dataset, future efforts will be directed toward fine-tuning the LLM.

Disclaimer: The videos sourced in this content are from YouTube and may be subject to licensing restrictions.

Additionally, we explored audio-based cues (Section 3: Audio Cue Analysis). While transcripts often highlight key moments, this approach was motivated by the assumption that variations in background noise, crowd reactions, and commentator tone could offer more precise temporal alignment. However, in practice, the results did not meet expectations, as the audio cues were noisy and inconsistent across the game.

Section 1 - Exploring RAG

RAG (Retrieval-Augmented Generation) is a technique that enhances language models by retrieving relevant external information and using it to generate more accurate, context-rich responses. It helps overcome limitations of fixed context windows. In the context of sports highlight generation, **RAG** assists in processing full game transcripts by first breaking them into manageable chunks and then identifying key events—such as goals, fouls, and substitutions within each chunk. These events are subsequently mapped to precise timestamps, to enable the generation of match highlights.

Our aim while generating match highlight through the transcript is to extract information about the following :

1. Event Detection – Find relevant events in lengthy, loosely structured text.
2. Temporal Precision – Accurately map events to timestamps.
3. Narrative Coherence – Ensure the final cut tells a clear and engaging story.

With this intent, we investigated three RAG-based approaches :

- Naive RAG - A straightforward method that retrieves segments using simple keyword- or score-based matching.
- Advanced Chunking for RAG - This technique divides the transcript into larger “parent” sections and smaller “child” chunks, enabling the model to retrieve both specific details and their broader context.
- Agentic RAG - A more sophisticated approach in which the model engages in multi-step reasoning to actively select and organize content for summarization.

Our investigation shows that while retrieval strategies such as **advanced chunking** (breaking transcripts into meaningful parts), the use of **vector databases** for semantic search, can enhance performance, the core challenge still persists. Precise event extraction and reliable **temporal anchoring** (tracking event timing across text) ultimately depend on improving the model’s intrinsic reasoning capabilities, a limitation that likely requires domain-specific fine-tuning.

The NAIVE Rag Approach

The core concept behind Naive RAG is to simplify the task of generating highlights by dividing it into two distinct roles: an information extractor and a final editor. The extractor's job is to analyze the full match transcript and identify every possible event goals, fouls, substitutions, and so on without considering the length of the final video. The editor's job is to take this exhaustive list of events and trim it down to fit within a strict 12 to 13-minute time limit.

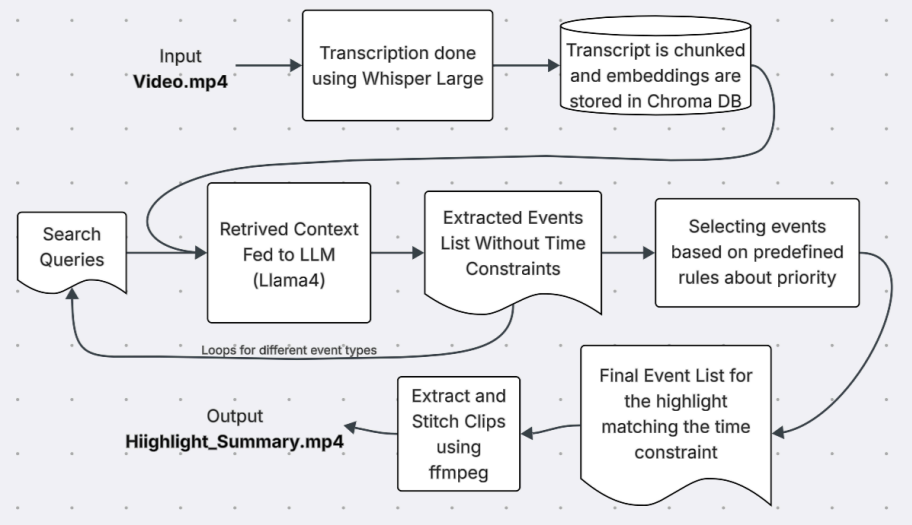

Fig 1 - The Naive RAG Architecture

The process began by breaking the long game transcript into smaller with overlapping segments (1000 tokens each). This was necessary because the model (Llama4:latest) couldn't process the entire transcript at once, and the overlap ensured continuity so that no important detail was cut off. These segments were then stored in a vector database, forming a searchable library indexed by semantic meaning.

During the extraction phase, the system queried the vector database for each event type starting with goals, then fouls with cards, substitutions, and so on. For each event type, the process is

made to retrieve matching segments and pass them to the model (Llama4:latest). As experiments showed , this led to a long, unfiltered list of moments.

In the final editing phase, a simple rule-based script filtered the events based on a defined priority order. High-priority events like goals were inserted first, followed by lower-priority events if the time constraints were satisfied . This predictable behavior met expectations; it ensured the final cut never exceeded the target duration and maintained a consistent structure across matches.

The final highlight reels lacked the flow and narrative coherence of a professionally edited summary. Most notably, the model (Llama4:latest) struggled with accurately mapping the timestamps to events. It frequently missed some goals entirely and identified the same event multiple times, causing repetitions in the final list. While the model performed well in identifying the high-level statistics, it was not reliable when it came to pinpointing the exact moments events occurred in the video.

These issues stemmed from two main limitations. First, the model only saw isolated segments during extraction, so it often missed the full context needed to understand when an event began and ended. A goal might be partially described across chunks, but if the model only received one of them, it could either miss it or create a fragmentary, incorrect timestamp. Second, the model lacked temporal anchoring; it had no inherent understanding of time progression across the transcript. It couldn't track that a particular buildup occurred five minutes before a goal or that a replay was being described again later, which led to duplicate detections and imprecise clips.

Example - Portugal vs Spain Match
Output : events.json

Advanced Chunking Strategy for RAG – Parent-Child Chunking Strategy

The idea behind this approach is to provide the model (Llama4:latest) with precision and context, allowing it to search for specific details while understanding the broader story surrounding them.

This is achieved through a key mechanism: **Parent document retrieval**. It allows for search by targeting specific text fragments but delivering the model the larger, complete paragraph they come from.

The approach begins by structuring the transcript into two levels: larger “parent” chunks that represent meaningful sections of the match, and smaller “child” chunks, each focused on a sentence or set of sentences. Only the child chunks are indexed for search, but each child retains a reference to its parent. This allows the model to retrieve with high specificity while still receiving rich, narrative context. Once relevant sentences are found, their parent chapters are retrieved to provide fuller context.

The final results expose clear limitations in the model’s reasoning ability. The model struggles with consistently identifying all goals and frequently produces repetitive results, often extracting the same event multiple times in slightly different forms. While it performs well in summarizing overall match narratives, it fails to reliably identify the exact timestamps and full context of events.

These shortcomings are due to two primary factors:
First, while the model receives richer input, it still lacks the reasoning depth to understand the full scope of a single event like tracking the buildup to a goal, the goal itself, and the reaction afterward as a single coherent moment.
Second, without a stable internal representation of time, it can’t consistently distinguish between similar textual fragments that refer to the same moment, leading to repeated or imprecise timestamping.

Example - Portugal vs Spain
Output : Output

Agentic RAG Approach

Unlike traditional Retrieval-Augmented Generation approaches, which retrieve relevant text and feed it to a single large language model, this multi-stage agentic architecture breaks down the complex timeline generation task into discrete, manageable steps. Each step is handled by a specialized agent, creating a modular and sequential flow that improves accuracy.

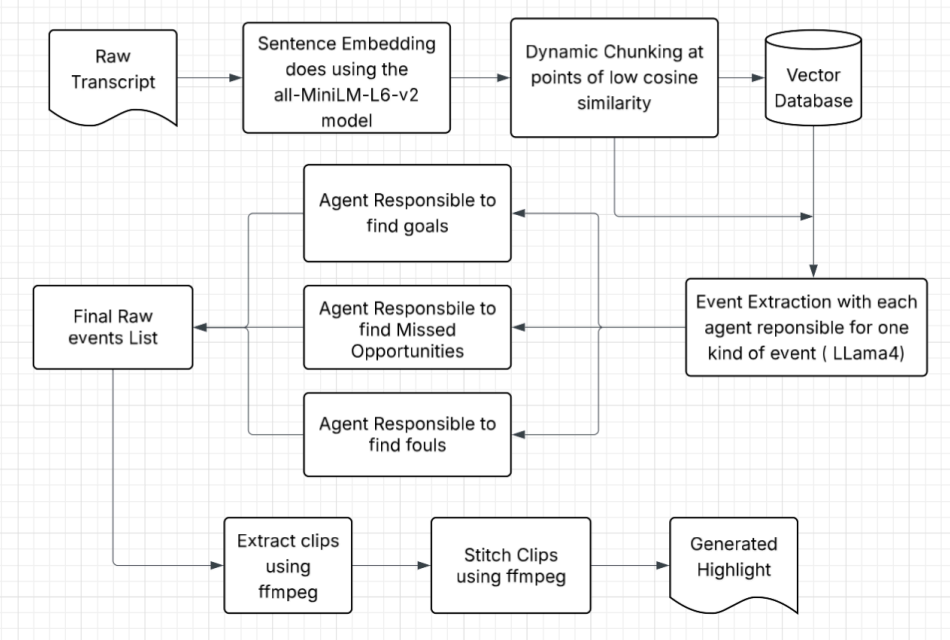

Fig 2 - Agentic Approach Depiction

The transcript is partitioned using a content-aware strategy, where divisions are based on semantic boundaries rather than fixed-length segmentation. Using sentence embeddings and similarity measures, natural topic boundaries are detected, and overlapping sentences are included to maintain context across chunk borders.

The core of this approach is the multi-agent event extraction stage. Instead of relying on a single, general-purpose prompt, the system employs specialized agents each focused on a narrow event type, such as goals, missed shots, or fouls with cards. Each agent uses a dedicated prompt tailored to its task and outputs structured JSON data.

This is done to reduce cognitive load on the model (Llama4:latest), increases precision, and improves the consistency of the extracted information. The agents operate independently of each other, producing a raw event list that may include some duplicates and minor inconsistencies due to overlapping input.

The model (Llama4:latest) struggled to identify all goals consistently and generated many repeated events. Although it excelled in summarizing overall narratives and determining final statistics such as the goal line, its ability to accurately map timestamps to events was poor. This was primarily due to the model's limited temporal reasoning and difficulties distinguishing closely related text fragments that referred to the same moment, resulting in losing or duplicated timestamps.

Ultimately, the Parent-Child RAG approach demonstrated that the infrastructure around the model had been pushed to its limits. The failure to achieve event-level precision was not due to flaws in the approach but rather the model's inability to reason deeply enough for the task. One of the viable paths forward is to improve the model itself through fine-tuning for this specific use case.

Example Run on the Portugal vs Spain Match - Output : Agentic Outputs

Section 2 - Dataset for Finetunning

This section introduces a curated collection of 100 soccer matches for training and fine-tuning the open-source Large Language Models (LLMs). Sourced from full-length match videos on YouTube. Each match is processed through an approach that transcribes the audio commentary using Whisher and then uses an API based generative model (Gemini) to create detailed event based timestamps in JSON format.

Dataset Curation Strategy

1. Playlist Ingestion: The approach begins by fetching all individual video URLs from a given YouTube playlist using yt-dlp.
2. Audio Extraction: For each video, the audio track is isolated and downloaded as a.wav file.
3. Transcription Generation: The extracted audio is transcribed using Whisper (large model). The transcription is segmented and timestamped, creating a precise text record of the match commentary linked to the video timeline (e.g., [01:15:32] And that's a brilliant goal!).
4. Generative Event Extraction (Gemini): The timestamped transcript is fed to a Google Gemini model with a prompt which instructs the model to act as an expert sports video editor and identify key narrative events. It extracts these events into a structured JSON format, creating a preliminary List of timestamps.
5. Structured JSON Output: The final output for each video is a single, self-contained JSON file (e.g., dataset_1.json, dataset_2.json) (Fig 2).

Dataset Format -

The dataset is organized into 10 sets, each representing a processing batch. The 100 files are named with a consistent prefix (dataset_) and a serial number, facilitating easy iteration.

Each JSON file in the dataset follows a consistent, and a structured schema:


```
{
    "sr_no": 1,
    "youtube_link": "https://www.youtube.com/watch?v=...",
    "audio_extracted": "Success",
    "transcript": "[00:00:05] Welcome to the grand final...\n[00:45:12] What a fantastic strike!...\n[...]",
    "events_data": {
        "edl_events": [
            {
                "start_time": "00:44:58",
                "end_time": "00:45:25",
                "team": "Team A",
                "type": "goal",
                "description": "Brilliant long-range strike"
            },
            {
                "start_time": "00:51:10",
                "end_time": "00:51:30",
                "team": "Team B",
                "type": "foul",
                "description": "Yellow card for a late challenge"
            }
        ],
        "match_analysis": {
            "match_summary": "An exciting match with Team A securing a late victory...",
            "teams": ["Team A", "Team B"],
            "match_duration": "01:45:30"
        },
        "total_events_found": 25,
        "processing_status": "success"
    }
}
```


Fig 3 . Dataset Format (JSON Format)

Key Features of the Dataset:

1. Temporally-Grounded Events: Every key event (goal, foul, missed goal, etc.) is tagged with a precise start_time and end_time. This allows models to learn the direct correlation between a text description and a specific segment of the video.
2. Narrative-Centric Structure: The prompt is designed to capture "atomic, self-contained mini-stories." A "goal" event, for example, includes the build-up play, the goal itself, the celebration, and the subsequent replays, as defined by its start_time and end_time.
3. Structured Metadata: Beyond events, each file contains:
    - Full Match Transcript: The complete, timestamped transcript is preserved for context or alternative training tasks.
    - High-Level Match Analysis: A generated match_summary, list of teams, and total match_duration provide a quick overview of the game.
    - Data Provenance and Quality: Fields like youtube_link ensure source traceability, while processing_status allows for easy filtering of any failed or problematic entries.
4. Categorized Event Types: Events are classified into a predefined set of types (goal, foul, missed goal, prologue, epilogue). This structured classification simplifies tasks like event counting, highlight reel generation (e.g., "create a reel of all goals"), and conditional analysis.

Link to the Dataset:

Batch No	Youtube Playlist Link	No. of Videos
Batch 1	https://youtube.com/playlist?list=PLW37-SNJcCyvmi8cp-8cuuZWEp_58dJcT&feature=shared	5
Batch 2	https://youtube.com/playlist?list=PLW37-SNJcCyu_IIJxjc0O_vD-7MbkwBFl&feature=shared	12
Batch 3	https://youtube.com/playlist?list=PLW37-SNJcCyt4kzd5nwLMANPH4qF9yylJ&feature=shared	13
Batch 4	https://youtube.com/playlist?list=PLW37-SNJcCysPpUkpJHPtBnw__nLFH9B3&feature=shared	9
Batch 5	https://youtube.com/playlist?list=PLW37-SNJcCytI5Ie2zrhSo2_mfQzBHT9U&feature=shared	11
Batch 6	https://youtube.com/playlist?list=PLW37-SNJcCyuKiGzPRvwvsrm0gxVTUeDW&feature=shared	13
Batch 7	https://youtube.com/playlist?list=PLW37-SNJcCytQ5vsL6xXFQVHmBt1SBOj8&feature=shared	10
Batch 8	https://www.youtube.com/playlist?list=PLoilDsc3ml0lNfkw-_1B3t1OvpkibX-O_	10
Batch 9	https://www.youtube.com/playlist?list=PLoilDsc3ml0lF2Ym4eaIrK9rLWWlCPfET	10
Batch 10	https://www.youtube.com/playlist?list=PLoilDsc3ml0k3NoGb7NW057kSSMxFDePA	8

Table 1. Playlists of the Datasets

Drive Link - Dataset_Finetuneing

Section 3 - Analysing Audio Cues for Sports Event Detection

Motivation

In sports broadcasting, key events are almost always accompanied by distinct audio cues: a spike in the commentator's voice, a roar from the crowd, or a specific sound like a basketball sneaker squeak or a cricket bat hit.

This Section explores analysis of three methodologies, starting with a simple threshold-based detection and advancing to more sophisticated, feature-weighted ranking systems using MFCCs. Audio from three different sports, Soccer (Portugal Vs Spain), Basketball (USAvSerbia), and Cricket (England Vs South Africa) are analyzed by separating commentator vocals from background noise to understand the unique contribution of each audio stream.

The Audio Analysis Approaches

Three approaches were tested. Each approach builds upon the last, introducing how "excitement" is defined and measured. The core audio features used are **Root Mean Square (RMS)** for energy/loudness, **Pitch** for vocal frequency, and **Mel-Frequency Cepstral Coefficients (MFCCs)** for sound timbre/quality.

1: Naive Threshold-Based Segmentation

This initial approach identifies segments where the audio signal is unusually loud or high-pitched. It provides a binary classification: a segment is either "excited" or "not."

- **Process:**
    1. **Feature Extraction:** Calculate RMS and Pitch from the audio signal over the clip in sections.
    2. **Thresholding:** A statistical threshold is set at the mean + 2 * standard deviations for both the smoothed RMS and Pitch values.

3. **Peak Identification:** Any section where either the RMS or Pitch exceeds its respective threshold is marked as a "peak time."
4. **Merging & Filtering:** Nearby peak times are merged into continuous segments, and short segments are discarded to reduce noise.

- **Outcome**: A list of all detected timestamps representing the events.

2: Ranked Intensity Scoring

This approach improves upon the first by not just identifying excited segments, but also by ranking them based on a composite score.

- **Process:**
    1. The same initial steps from Approach 1 are used to define the segments.
    2. **Scoring:** For each valid segment, an Intensity Score is calculated by summing the average RMS and the average Pitch (Intensity = avg_RMS + avg_Pitch).
    3. **Ranking:** Segments are sorted in descending order based on their Intensity Score.
- **Outcome**: A ranked list of the top-k most excited segments, providing a hierarchy of importance.

3: MFCC-Enhanced Scoring

This approach incorporates MFCCs to capture the timbral quality of the sound, which helps differentiate between various types of audio events beyond simple loudness or pitch.

- **Process**:
    1. In addition to RMS and Pitch, MFCCs are extracted from the audio.
    2. A weighted score is calculated for each segment using the formula:
    Intensity = (0.4 * avg_RMS) + (0.3 * avg_Pitch) + (0.3 * avg_MFCC_Energy)

3. **Ranking**: Segments are sorted based on the new intensity score.

- **Outcome**: A ranked list of top-k segments where the ranking reflects a combination of loudness, pitch, and timbral change, offering a more robust measure of a significant audio event.

Experimental Results

The following tables summarize the results from applying the three approaches to the complete set of audio files.

Naive Approach - All Detected Excited Segments

Clip Name	Type	Timestamps
Portugal vs Spain .mp3 (Soccer)	Vocal	22.62s to 23.38s, 26.26s to 26.89s
	BG	16.07s to 17.90s
USAvSerbia (Basketball)	Vocal	[List of 42 segments, e.g., 4.13s to 7.78s, ...]
	BG	[List of 24 segments, e.g., 15.28s to 16.83s, ...]
England Vs South Africa (Cricket)	Vocal	[List of 45 segments, e.g., 0.88s to 1.76s, ...]
	BG	[List of 25 segments, e.g., 2.04s to 2.62s, ...]

Table 2. Naive Approach - All Detected Excited Segments

Observation: The naive approach is highly sensitive, generating a large number of segments for both vocal and background tracks. This flood of data makes it difficult to isolate truly critical actual events without a more intelligent ranking system.

Approach 2 - Top Ranked Segments (RMS + Pitch)

Clip Name	Type	Top Excited Segments
Portugal vs Spain (Soccer)	BG	1. 17.14s to 17.93s | Score: 949.4431
USAvSerbia (Basketball)	Vocal	1. 241.79s to 242.39s | Score: 683.9261
	BG	1. 37.22s to 38.17s | Score: 2717.0767 (... and 9 others)
England Vs South Africa (Cricket)	Vocal	No segments in top ranking
	BG	1. 159.36s to 159.94s | Score: 1659.0967 2. 143.06s to 145.12s | Score: 719.0846 (... and 4 others)

Table 3. Top Ranked Segments Approach

Observation: Ranking effectively prioritizes events. In this, the highest-scoring segments come from the background (crowd) audio, suggesting that massive crowd reactions were the most intense audio events. Notably, no vocal segments from the cricket match made the top-ranked list, indicating that the crowd roared at key moments.

Approach 3 - Top Segments (MFCC-Enhanced Scoring)

Clip Name	Type	Top Excited Segments (MFCC-enhanced scoring)
Portugal vs Spain (Soccer)	BG	1. 17.14s to 17.93s | Score: 293.2683
USAvSerbia (Basketball)	Vocal	1. 241.79s to 242.39s | Score: 213.7909
	BG	1. 37.22s to 38.17s | Score: 825.7901 (... and 10 others)
England Vs South Africa (Cricket)	Vocal	No segments in top ranking
	BG	1. 159.36s to 159.94s | Score: 507.2436 2. 143.06s to 145.12s | Score: 230.2210 (... and 4 others)

Table 4 . Top Ranked Segments Approach MFCC Based

Observation: The relative ranking of the top moments is preserved from Approach 2, demonstrating consistency. The inclusion of MFCCs provides a more balanced score, less likely to be skewed by a single feature like a sudden, loud noise.

Sport-Specific Analysis and Challenges

The nature of each sport creates a unique audio landscape, presenting different observations and challenges for analysis.

Sport	Observation	Challenges
Basketball	• Frequent, distinct events (baskets, fouls) lead to numerous, sharp audio cues. • Commentator speech is often direct and consistently paced between plays.	• High Event Frequency: The model is over-sensitive, requiring effective ranking to find true highlights. • Overlapping Sounds: Sneakers, ball, buzzers, and crowd noise often overlap, making source separation difficult.
Cricket	• The distinct "bat on ball" sound is a powerful audio cue. • Crowd reactions are often massive and directly tied to key events (wickets, sixes), making the BG track an indicator, as confirmed by the results.	• Dominant Crowd Noise: As seen in the results, the crowd's roar can completely overshadow the commentator's excitement, making the vocal track less useful for top-tier event detection in some cases. • Misleading Cheers: The crowd may cheer for extended periods not tied to a single, climactic event, potentially creating long, low-intensity segments.
Soccer	•In these games, commentators build excitement vocally, providing a clear ramp-up to a key moment.	• Mixed Voices: Crowd and commentator voices can blend, making clean separation challenging. • Constant Noise: An active crowd can create a high level of "white noise," raising the baseline and making it harder for a static threshold to detect event-driven roars.

Table 5. Sport Wise Analysis

Overall Challenges

This study highlighted several overarching challenges inherent in this type of audio analysis:

1. **Threshold Sensitivity:** The use of a static threshold (mean + 2*std) is a primary limitation. It may be too high for quiet games or too low for loud ones. An adaptive thresholding system is a clear next step.
2. **Feature Weighting:** The weights in Approach 3 were chosen heuristically. These likely need to be tuned specifically for each sport to optimize performance.
3. **Lack of Context:** These models detect "excitement" but have no semantic understanding. A commentator shouting about a controversial call is indistinguishable from one shouting about a goal.
4. **Audio Quality & Separation:** The effectiveness of the analysis depends heavily on the quality of the source audio and the ability to cleanly separate vocal and background tracks.

Audio artifacts of the Soccer Match :

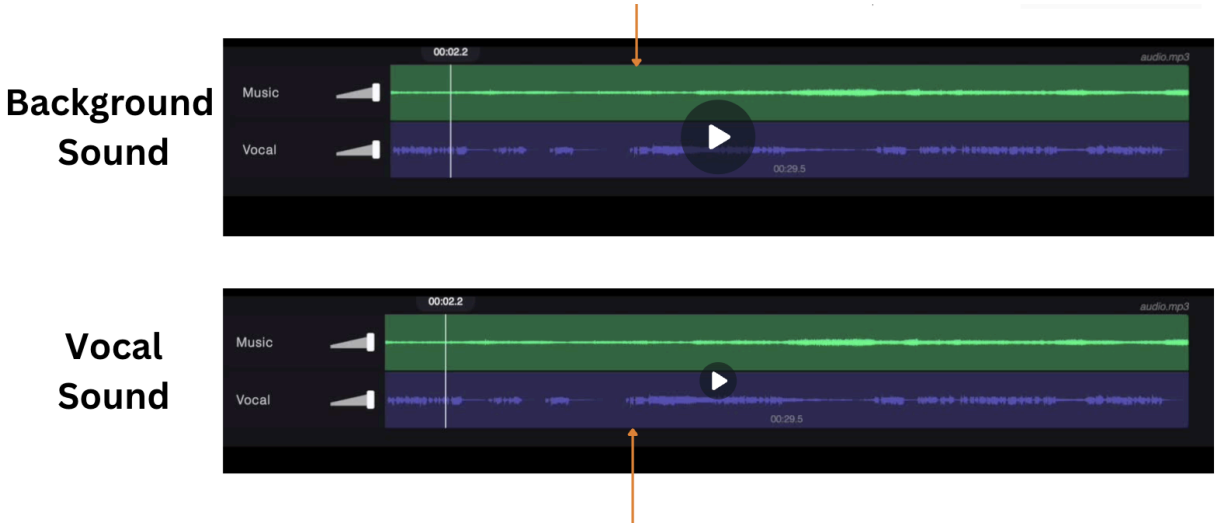

Fig 4 . Background and Vocal Sound of Portugal Vs Spain 2018 match

In the above Figure , the orange arrow represents the actual goal event (at 9.01 sec) in the given clip.

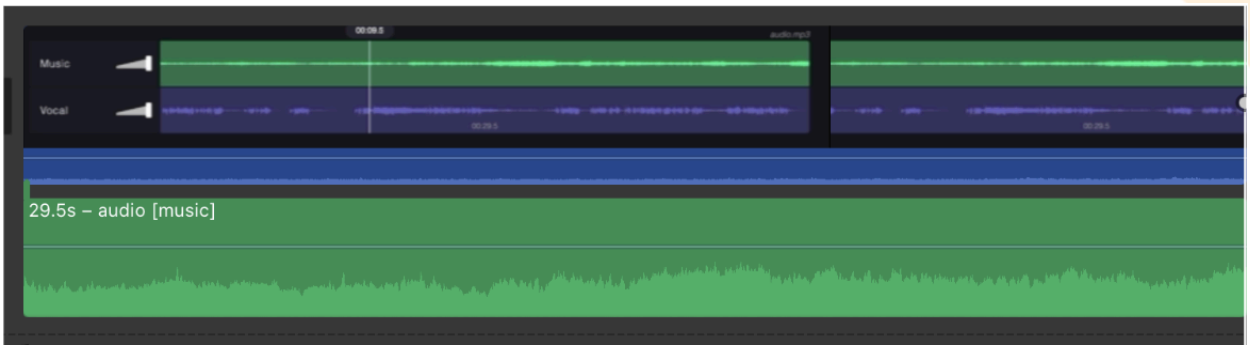

Fig 5. Audio waveform of Portugal Vs Spain 2018 match commentary.

Observation based on the Audio sample:

- The Commenter's Vocal and Crowd background is blended in the Broadcast.
- The background is White Noised and Consistent.
- The Commenter’s tone (Pitch and Amplitude) is mixed with Background Noise.

Conclusion and Future work:

This analysis of highlight generation using the open-source LLM, Llama4, revealed that while Retrieval-Augmented Generation (RAG) can address context window limitations, it ultimately fails due to the model's core inability to perform robust temporal reasoning and synthesize disparate events.

This finding resulted in a strategic pivot from retrieval-centric workarounds to fundamentally enhancing the model's capabilities through domain-specific fine-tuning, enabled by the creation of a comprehensive 100-match annotated dataset.

Hence the future work will focus on finetuning the Llama4 model.

Additionally, it was observed that audio analysis is constrained by persistent background noise from the crowd, making it difficult to reliably isolate the distinct audio spikes that signal key events.