**Rayv Code-o-Tron 3000 Hackathon Essay Submission**

**Title:** Multimodal Deep Learning for Safe Content Classification in Short Videos

**Introduction**

With the rise of short-form video platforms like TikTok, the challenge of detecting harmful or unsafe content in real-time has become increasingly important. As a beginner, I approached this hackathon with the aim of exploring how AI and computer vision can be used to solve this real-world problem using both classic and advanced deep learning methods. This journey taught me the importance of modular model design, dataset preparation, and fusion of different data types (visual, audio, and text).

**Dataset Challenges**

I used the **TikHarm Dataset**, which is 30 GB+ in size and structured similarly to UCF101. It contains labeled videos under four categories: Safe, Adult, Harmful, and Suicide. Each class varies in content complexity and duration, and the dataset includes separate folders for train, test, and validation.

**Models Explored in Phases**

**Phase 1: Frame-Based CNN (MobileNetV2)**

* Extracted 1 FPS frames from videos.
* Applied MobileNetV2 on each frame and used majority voting.
* **Pros:** Fast, great baseline.
* **Cons:** Ignores motion or audio cues.

**Phase 2: Metadata-Based Classical ML (Random Forest)**

* Used duration, category, and synthetic notes.
* Applied TF-IDF + OneHot encoding.
* **Pros:** Quick to implement.
* **Cons:** Doesn’t use actual video or sound.

**Phase 3: Modal-Level Models**

* Separate models for text (BERT), visual (CNN), and audio (MFCC + SVM).
* Helped understand each modality's strength.

**Phase 4: CNN + LSTM for Sequences**

* Used MobileNetV2 + LSTM on a sequence of frames.
* Captured temporal patterns (e.g., buildup of aggression).
* **Challenge:** Heavy on GPU, complex to debug.

**Phase 5: 3D CNN (R3D18)**

* Input video as a 3D tensor (C, T, H, W).
* Fine-tuned pretrained r3d\_18 model.
* **Why it’s one of the best:**
  + Captures both **what** is happening and **how it evolves**.
  + Excellent for detecting motion-heavy scenes like fights, abuse, etc.
* **Issues faced:**
  + GPU memory.
  + Preprocessing required consistent frame shapes and lengths.

**Phase 6: Multimodal Fusion Model**

* Combined predictions from audio, visual, and text using Logistic Regression.
* Audio: MFCCs from librosa.
* Visual: Mocked confidence (can be MobileNet or ViT).
* Text: TF-IDF on notes.
* **Why it's best for hackathon scalability:**
  + Lightweight and interpretable.
  + Allows fallback if one modality is missing.
  + Easy to improve by upgrading base models.

**Phase 7: Vision Transformer (ViT)**

* Tokenized frames using ViTFeatureExtractor.
* Averaged softmax outputs across frames.
* **Result:** SOTA accuracy on image-based scenes but lacks audio/text context.

**Why Phase 5 and 6 Are the Best**

✅ **3D CNN (Phase 5)** is best when your compute budget allows and you want a deep, video-only model that "understands" motion.

✅ **Fusion Model (Phase 6)** is best for hackathon deployment because:

* It uses all three modalities.
* You can easily swap or improve models.
* It generates explainable outputs: label, confidence, and notes.

**Combining both is ideal**: Use 3D CNN as visual backbone in a future fusion model.

**Conclusion**

I explored AI models through this hackathon — from basic classical ML to advanced 3D CNNs and Transformers. As a beginner, building and debugging multimodal pipelines was tough but rewarding. I now understand how crucial it is to align models with both the dataset and real-world constraints. My final system fuses visual, audio, and text content to give robust, confident predictions, making it scalable and effective for content moderation platforms.

**Thank you for this opportunity!**