بسم الله الرحمن الرحیم

گزارش کار پروژه

عنوان: Music-to-Video-Generator

درس: سیستم‌های نهفته و بی‌درنگ

استاد: مهندس مهدی سیفی‌پور

اعضای تیم: سید مهدی منجم، علیرضا میرزایی، محمدحسین فرهادیان

تابستان ۱۴۰۴

چکیده

این پروژه با هدف **تبدیل موسیقی به ویدئو با استفاده از هوش مصنوعی** انجام شد. ابتدا فایل صوتی به فرمت مناسب تبدیل شد و متن آن با استفاده از مدل Whisper استخراج شد. سپس، با بهره‌گیری از **مدل Gemini 2.0 Flash** یک پرامپت سینمایی از متن تولید شد. این پرامپت به مدل‌های **Stable Diffusion XL** و **Stable Video Diffusion** داده شد تا ابتدا تصویر اولیه و سپس ویدئوی نهایی تولید شود. نتایج نشان می‌دهد که این روش می‌تواند به صورت **خودکار و با کیفیت مناسب** موسیقی را به محتوای بصری تبدیل کند.

مراحل اجرای پروژه

گام ۱: استخراج متن از موسیقی

- فایل ورودی موسیقی (MP3) به WAV تبدیل شد تا مدل **Whisper** بتواند آن را پردازش کند.
- با استفاده از Whisper، متن موسیقی استخراج و در فایل `lyrics.txt` ذخیره شد.
- این مرحله پایه‌ای برای تولید پرامپت سینمایی و در نهایت تصویر و ویدئو بود.

گام ۲: تولید پرامپت سینمایی با Gemini

- متن استخراج شده توسط Whisper به مدل **Gemini 2.0 Flash** داده شد.
- مدل یک پرامپت کوتاه، خلاقانه و توصیفی تولید کرد که شامل **فضای بصری**، **رنگ‌ها**، **احساسات و محیط** بود.
- پرامپت تولید شده در فایل `video_prompt.txt` ذخیره شد و برای مراحل بعدی استفاده شد.

گام ۳: تولید تصویر اولیه باStable Diffusion XL

- پرامپت سینمایی به مدل **Stable Diffusion XL** داده شد.
- تصویر اولیه با کیفیت بالا تولید و در مسیر پروژه ذخیره شد.(`initial_image.png`)
- این تصویر به عنوان پایه برای ساخت ویدئو در مرحله بعد عمل کرد.

گام ٤: تولید ویدئو باStable Video Diffusion

- تصویر اولیه به مدل **Stable Video Diffusion** داده شد.
- ویدئو با طول ۲۵ فریم تولید و با نرخ فریم مناسب ذخیره شد.(`generated_video.mp4`)
- از تکنیک‌های کاهش رزولوشن و مدیریت حافظه برای اجرا در محیط Colab استفاده شد.

نتایج و تحلیل

- تولید تصویر و ویدئو با کیفیت مناسب انجام شد(نسبت به فضای پردازشی در دسترس) و ارتباط بین موسیقی و عناصر بصری حفظ شد.
- استفاده از پرامپت خلاقانه باعث شد **حس و رنگ موسیقی در تصویر و ویدئو منتقل شود.**
- محدودیت‌های مدل‌ها و حافظه GPU باعث شد که برای ویدئوهای طولانی‌تر یا رزولوشن بالا نیاز به تنظیمات اضافی باشد.

۵ .چالش‌ها و نحوه حل آن‌ها

- **فضای VRAM محدود :** مدل‌های تولید ویدئو نیاز به حافظه بالایی داشتند. برای حل مشکل، رزولوشن تصویر و ویدئو کاهش یافت و نرخ فریم (fps) پایین آمد تا حافظه GPU کافی باشد. همچنین کارکردهای پردازشی تا حد ممکن در بین سلولهای مختلف کد توزیع شد تا بار پردازشی تا حد ممکن توزیع شده باشد.
- **پاسخ ندادن API Gemini :** در برخی موارد، پرامپت به دلیل محدودیت‌های سرور پاسخ نمی‌گرفت. برای این مشکل از تکرار درخواست و مدیریت خطا استفاده شد تا برنامه بدون دلیل متوقف نشود.
- **مدیریت حجم داده‌ها :**برای جلوگیری از مصرف بیش از حد حافظه و کاهش خطر کرش شدن Colab ، تصاویر و فریم‌ها به صورت مرحله‌ای پردازش و ذخیره شدند.

٦ .جمع‌بندی

این پروژه نشان داد که **در صورت دسترسی به منابع پردازشی مناسب**، می‌توان با **ترکیب مدل‌های تبدیل گفتار به متن، تولید محتوا و تولید تصویر/ویدئو** یک سیستم خودکار برای تولید محتوای بصری از موسیقی ایجاد کرد. مراحل پروژه به صورت ماژولار طراحی شده‌اند و هر بخش می‌تواند **به صورت جداگانه بهبود یافته یا جایگزین مدل‌های جدید شود.**

مدل‌های مورد استفاده

۱. Whisper: https://github.com/openai/whisper
۲. Stable Diffusion XL: https://stability.ai
۳. Stable Video Diffusion: https://stability.ai/blog/stable-video-diffusion
۴. Gemini 2.0 Flash API: https://www.gemini.com