<https://colab.research.google.com/drive/14tRtsnY4bsWWnl4H73CTxdc8jIMmCQTJ?usp=sharing>

<https://g.co/gemini/share/f755e29f12fd>

Dünkü denemelerde yaşadığım veri işleme sorunlarının front-end tarafından gönderilen ses formatından kaynaklanabileceğini düşündüğüm için, bugün bu hipotezi test etmek amacıyla kendi test ortamımı kurmaya karar verdim. İlk olarak Google Colab üzerinde basit bir HTML arayüzü tasarladım ve ngrok aracılığıyla bu arayüze erişim sağlayan geçici bir web sitesi oluşturdum. Bu kontrollü ortamda yaptığım denemelerde sistemim herhangi bir sorun çıkarmadan çalıştı. Hatta Colab'in sağladığı GPU gücü sayesinde, bazı anlık çıktılar hatalı olsa da, gerçek zamanlı transkripsiyon yapabildiğimi gördüm. Bu test, kendi mimarimin temel olarak doğru çalıştığını kanıtlamış oldu.

Bu başarılı denemenin ardından Colab üzerindeki kodu kendi bilgisayarıma uyarlayarak front-end ekibiyle tekrar bir entegrasyon denemesi yaptık, fakat yine başarılı olamadık. Hatanın kökenini araştırdığımızda, front-end tarafının WebSocket üzerinden çok yoğun bir şekilde ham binary paketler gönderdiğini tespit ettik. Normalde mimarim, gelen ham PCM verilerini bir byte dizisinde biriktirip belirli bir boyuta ulaştığında bir NumPy dizisine dönüştürerek modele gönderecek şekilde tasarlanmıştı. Ancak front-end, veriyi biriktirmeden tek tek yolladığı için sistemim aşırı veri yüklemesiyle karşılaşıyor, şişiyor ve bir süre sonra WebSocket bağlantısını sonlandırıyordu.

Front-end ekibi bu hatayı düzelttikten sonra yeni bir sorunla karşılaştık. Bu kez front-end tarafından gönderilen 1 saniyelik bir ses kaydı, benim sistemim tarafından 3-4 saniyelik bir ses gibi algılanıyordu. Yaptığımız inceleme sonucunda bu sorunun, sesin örnekleme oranından (sampling rate) kaynaklandığını anladık. Modelim 16.000 Hz frekansında çalışmaya ayarlı iken, front-end tarafı sesi bu frekansa düşürmeden, 64.000 Hz gibi çok daha yüksek bir frekansta gönderiyordu. Front-end ekibi bu sorunu çözmek üzere çalışmalara başladı.

ÖZET => Colab ve ngrok ile kurulan test ortamında sistemin çalıştığı doğrulandı, front-end ile entegrasyonda ise önce aşırı paket gönderimi, ardından da yanlış örnekleme oranı (sampling rate) sorunları tespit edilerek çözülmeye başlandı.

16.Gün ÖZET => Colab üzerinde ngrok ile test edilen bir prototip geliştirildi, ardından mühendis geri bildirimiyle WebSocket'e geçiş için FastAPI tabanlı yeni bir servis yazıldı ve front-end'den gelen blob verisi işlendi.

17.Gün ÖZET => FastAPI ve WebSocket servisi ayağa kaldırıldı, ancak FFmpeg'in .webm ses paketlerini .wav formatına dönüştürmede yavaş kalması nedeniyle gerçek zamanlı transkripsiyonda veri kaybı yaşandı.

18.Gün ÖZET => Colab ve ngrok ile kurulan test ortamında sistemin çalıştığı doğrulandı, front-end ile entegrasyonda ise önce aşırı paket gönderimi, ardından da yanlış örnekleme oranı (sampling rate) sorunları tespit edilerek çözülmeye başlandı.