# 🚀 NVIDIA Riva Speech AI 서버 배포 및 활용 가이드 (Docker 기반)

이 문서는 NVIDIA Riva Speech AI 서버를 Docker 환경에서 설치, 실행, 그리고 Python을 통해 호출하는 과정을 단계별로 정리한 레퍼런스 가이드입니다.

## 1단계. 환경 준비 및 필수 도구 확인 (Prerequisites)

Riva 서버 구동을 위해 필요한 기본 환경 및 도구의 설치 여부를 확인합니다.

### 1.1. Docker 설치 확인

docker --version  
docker-compose --version

* **필수:** Docker Desktop (Windows/macOS) 또는 Linux용 Docker가 설치되어 있어야 합니다.

### 1.2. GPU 환경 설정 (NVIDIA GPU 사용 시)

Riva는 GPU 가속을 사용하므로, NVIDIA GPU를 사용하는 환경이라면 반드시 **NVIDIA Container Toolkit**(nvidia-docker2)이 설치되어 있어야 합니다.

* **확인:** docker run --gpus all nvidia/cuda:11.8.0-base-ubuntu22.04 nvidia-smi 등의 명령어로 GPU가 Docker 컨테이너 내에서 인식되는지 테스트해 보세요.

### 1.3. NGC 로그인 (모델 다운로드 시)

NVIDIA NGC (NVIDIA GPU Cloud)에서 Riva 모델을 다운로드하기 위해 로그인이 필요합니다.

docker login nvcr.io

* **요구 사항:** 유효한 NVIDIA NGC 계정과 API 키가 필요합니다.

## 2단계. Riva 설정 파일 (config.sh) 준비

Riva 초기화 스크립트 실행 전, 서버가 어떤 서비스와 모델을 사용할지 정의합니다. 아래는 영어(en-US) 환경에서 TTS, ASR, NLP를 모두 활성화하는 설정입니다.

### 2.1. 핵심 설정 (config.sh 반영 내용)

| **설정 변수** | **값** | **설명** |
| --- | --- | --- |
| service\_enabled\_asr | true | 자동 음성 인식 (ASR) 서비스 활성화 |
| service\_enabled\_nlp | true | 자연어 처리 (NLP) 서비스 활성화 |
| service\_enabled\_tts | true | 텍스트-음성 변환 (TTS) 서비스 활성화 |
| language\_code | ("en-US") | 사용할 언어를 영어(미국)로 지정 |
| riva\_model\_loc | "/workspace/assessment\_model\_repo" | 모델 파일 (RMIRs 및 최적화된 모델)을 저장할 경로 지정 |
| use\_existing\_rmirs | false | riva\_init.sh 실행 시 NGC에서 RMIR을 **새로 다운로드**하도록 지정 |

## 3단계. 모델 다운로드 및 배포 준비 (riva\_init.sh)

준비된 config.sh를 바탕으로 필요한 모델을 NGC에서 다운로드하고, Riva 서버에서 사용 가능한 최적화된 형식으로 배포 준비를 합니다.

### 3.1. 초기화 스크립트 실행

./riva\_init.sh

* **작동 원리:** 이 스크립트가 riva\_model\_loc에 지정된 경로에 모델 아티팩트 (RMIRs)를 다운로드하고, 이들을 Riva 서버용으로 최적화하여 준비합니다.
* **주의:** 실행 중 NGC API Key 입력 및 EULA 동의가 필요할 수 있습니다.

## 4단계. Riva 서버 Docker 컨테이너 실행

준비된 모델을 마운트하고 Riva 서버를 실행합니다.

### 4.1. 서버 실행 명령어 (GPU 환경 기준)

docker run -it --rm --gpus all \  
 -p 50051:50051 \  
 --name riva-speech \  
 -v /workspace/assessment\_model\_repo:/riva-model-repo \  
 nvcr.io/nvidia/riva/riva-speech:2.8.1

| **옵션** | **설명** |
| --- | --- |
| --gpus all | 호스트 시스템의 모든 GPU를 컨테이너에 할당 |
| -p 50051:50051 | 호스트 포트 50051을 컨테이너 포트 50051 (Riva gRPC 기본 포트)로 매핑 |
| --name riva-speech | 컨테이너 이름을 riva-speech로 지정 |
| -v <Host Path>:<Container Path> | 호스트의 모델 저장소(riva\_model\_loc)를 컨테이너 내부(/riva-model-repo)에 마운트 |
| nvcr.io/...:2.8.1 | 사용할 Riva Speech API Docker 이미지 |
| -it --rm | 컨테이너를 상호 작용 모드로 실행하고 종료 시 자동으로 제거 |

### 4.2. CPU 환경에서 실행 시 (선택 사항)

GPU를 사용하지 않는 환경에서는 --gpus all 옵션을 제거하고 실행합니다. (단, 이 경우 성능이 크게 저하됩니다.)

## 5단계. Riva 서버 상태 확인 및 Python 테스트

서버 컨테이너가 성공적으로 구동되었는지 확인하고, 클라이언트 코드에서 서비스를 호출하여 테스트합니다.

### 5.1. 컨테이너 상태 확인

실행 중인 컨테이너 목록과 상태를 확인합니다.

docker ps

### 5.2. 서버 로그 확인

서버 시작 과정이나 오류를 실시간으로 확인합니다.

docker logs -f riva-speech