

# 메타버스 내 익명 화자 인식 시스템 구현

## 1. 연구 목적, 배경 및 contribution

### 배경:

- 메타버스 플랫폼(VRChat, Roblox 등)에서 익명성을 악용한 모욕, 성범죄, 사기 사례 증가
- 현재는 문제 사용자가 계정을 변경하면 신원 추적이 불가능한 상황
- 대부분의 범죄가 1:1 환경에서 발생하여 증인이 없고 증거 수집이 어려움

### 목적:

- 익명 환경에서 문제 사용자의 음성 특징을 기반으로 동일인 여부를 식별하는 시스템 개발
- 사용자 평가 시스템과 화자 인식 기술을 결합한 실시간 위험 사용자 경고 시스템 구현

### Contribution:

- 메타버스 환경에 특화된 화자 인식 응용 방안 제시
- 익명 평가와 화자 인식을 결합한 새로운 사용자 보호 메커니즘 제안
- 오픈소스 화자 인식 툴(ESPnet)의 메타버스 적용 사례 연구

## 2. 수행 계획 및 방법

### 1단계: 사용자 평가 시스템 설계 (2주)

- 메타버스 내 "싫어요" 버튼 또는 익명 평가 인터페이스 설계
- 평가 데이터 저장 및 관리 DB 구조 설계

### 2단계: 음성 수집 및 특징 추출 (3주)

- 평가가 낮은 사용자의 음성 샘플 수집 모듈 개발
- ESPnet을 활용한 화자 임베딩(speaker embedding) 추출

python

```
# ESPnet을 활용한 화자 임베딩 추출 간략 코드
import torch
from espnet2.bin.spk_embedding import SpeakerEmbedding

model = SpeakerEmbedding("espnet/speaker_embedding_model")
audio, sr = librosa.load("problem_user.wav", sr=16000)
emb = model.embed(torch.tensor(audio))
```

### 3단계: 화자 인식 모델 구현 (3주)

- 화자 임베딩 간 유사도 계산 알고리즘 구현
- 임계값(threshold) 설정 및 동일 화자 판별 로직 구현

python

```
# 화자 유사도 계산 및 판별 간략 코드
def compare_speakers(emb1, emb2, threshold=0.7):
    similarity = cosine_similarity(emb1.reshape(1, -1), emb2.reshape(1, -1))[0][0]
    return similarity > threshold, similarity
```

### 4단계: 통합 시스템 개발 (3주)

- 새로운 사용자 접속 시 기존 DB와 실시간 비교 시스템 구현
- 특정 임계값 이상 유사 시 경고 알림 기능 구현
- 간단한 웹/앱 인터페이스 개발

### 5단계: 테스트 및 성능 평가 (3주)

- 실제 메타버스 환경에서 PoC 테스트 진행
- 동일 화자 식별 정확도 측정 및 개선점 도출

### 3. 기술 선택 이유

#### ESPnet 선택 이유:

- 오픈소스 프로젝트로 접근성이 높고 커스터마이징 용이
- 짧은 음성 샘플(5-10초)로도 화자 특성 추출 가능
- 다양한 음향 환경에서의 화자 인식 성능이 검증됨
- PyTorch 기반으로 학습 및 추론 과정 수정이 용이함

#### 익명 평가 시스템 도입 이유:

- 커뮤니티 자정 작용을 통한 문제 사용자 초기 식별 가능
- 무분별한 음성 수집 방지 및 프라이버시 침해 최소화
- 평가가 낮은 사용자에게 대해서만 선별적 모니터링으로 시스템 부하 감소

#### 코사인 유사도 적용 이유:

- 화자 임베딩 벡터 간 유사도 측정에 널리 사용되는 표준 방식
- 계산 복잡도가 낮아 실시간 비교에 적합
- 0-1 사이의 정규화된 값으로 임계값 설정이 직관적

### 4. 기대 output

#### 기술적 산출물:

- ESPnet 기반 메타버스 환경 최적화 화자 인식 모듈
- 익명 평가 및 화자 인식 연동 시스템 프로토타입
- 화자 임베딩 DB 및 관리 시스템

#### 구현 가능한 기능:

- 저평가 사용자의 음성 특징 자동 저장
- 신규 계정 접속 시 기존 저평가 사용자와의 유사도 계산
- 유사도 80% 이상 시 해당 사용자에게 경고 메시지 발송

#### 성능 목표:

- 동일 화자 식별 정확도 75% 이상 (PoC 수준)

- 오탐지율(False Alarm Rate) 20% 이하
- 실시간 처리 가능한 수준의 지연시간(1초 이내)

**활용 방안:**

- VRChat 및 Roblox와 같은 메타버스 플랫폼의 모듈형 플러그인으로 제공
- 메타버스 내 안전한 커뮤니티 형성을 위한 보조 도구로 활용
- 향후 메타버스 플랫폼의 사용자 보호 정책 수립에 기여