

## 과제 #3: Integrating C# code in Unity with Neural Net code in Python

전자공학과 120220117 박준호

### 1. 목적

- ✓ C# 기반의 unity 프로그래밍에서 python 기반의 심층 신경망 모델을 구현하기 위함

### 2. Python.NET 개요

- A. Python.NET(pythondotnet)은 Windows, Linux, 그리고 macOS에서 .NET 프레임워크, .NET Core 및 mono 런타임과 거의 완벽하게 통합되도록 하는 패키지
- B. Python.NET을 사용하면 CLR(C#, VB)을 대상으로 하는 모든 언어로 작성된 .NET 서비스 및 구성 요소를 사용하여 .NET 어플리케이션을 코딩하거나 python으로 전체 어플리케이션을 빌드할 수 있음

### 3. 구현 과정

- A. 로컬에 설치한 python 가상환경에 있는 pythonXX.dll 파일 경로 파악
- B. 가상환경 경로 및 python 코드가 저장되어 있는 디렉토리 경로 파악
- C. 아래와 같이 python 작업 폴더 및 가상환경을 불러오는 C# 코드 작성

```
public class GesticulatorDemo : MonoBehaviour
{
    void Start()
    {
        Main();
    }

    void Update()
    {
    }

    public static void AddEnvPath(params string[] paths)
    {
        var envPaths = Environment.GetEnvironmentVariable("PATH").Split(Path.PathSeparator).ToList();
        envPaths.InsertRange(0, paths.Where(x => x.Length > 0 && !envPaths.Contains(x)).ToArray());
        Environment.SetEnvironmentVariable("PATH", string.Join(Path.PathSeparator.ToString(), envPaths), EnvironmentVariableTarget.Process);
    }

    static void Main()
    {
        Runtime.PythonDLL = @"C:\Users\VDL\anaconda3\envs\gesti\python37.dll";
        var PYTHON_HOME = Environment.ExpandEnvironmentVariables(@"C:\Users\VDL\anaconda3\envs\gesti");
        AddEnvPath(PYTHON_HOME, Path.Combine(PYTHON_HOME, @"Library\bin"));
        PythonEngine.PythonHome = PYTHON_HOME;
        PythonEngine.PythonPath = string.Join(
            Path.PathSeparator.ToString(),
            new string[]
            {
                PythonEngine.PythonPath,
                Path.Combine(PYTHON_HOME, @"Lib\site-packages"),
                @"C:\Users\VDL\Desktop\github\metaverse_hw3\Assets"
            }
        );
        PythonEngine.Initialize();
        using (Py.GIL())
        {
            dynamic demo = Py.Import("demo.demo");
        }
        PythonEngine.Shutdown();
    }
}
```

### 4. 예상 구현 결과

- A. SMPL과 다른 format으로 이루어진 모델을 SMPL 모델로 retargeting 하도록 학습된 심층 신경망 모델에서부터 파라미터를 불러온 후, 이를 unity 상의 다양한 parameter로 사용하게 하여 input(text, audio)을 넣었을 때 원하는 output이 나오도록 함