동양미래대학교
DONGYANG MIRAE UNIVERSITY

KDD2024
BARCELONA, SPAIN
Workshop on AI for Education (AI4EDU)

ChatGPT and Its Educational Impact: Insights from a Software Development Competition

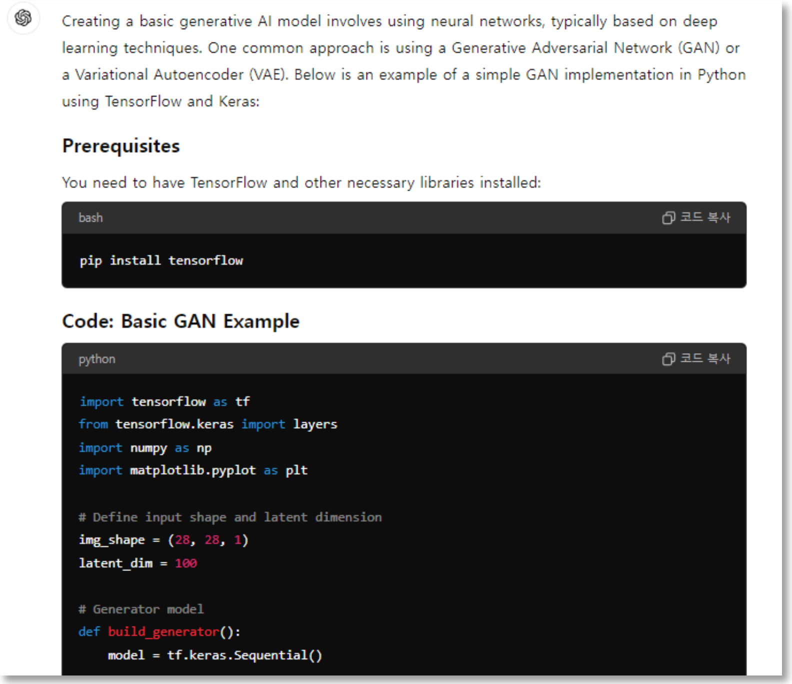
Sunhee Hwang, Yoodoo Kim, Heejin Lee
Dongyang Mirae University

Generative AI in Software Development

- Generative AI is reshaping the software development landscape

Generative AI in Software Development

Code Generation



Creating a basic generative AI model involves using neural networks, typically based on deep learning techniques. One common approach is using a Generative Adversarial Network (GAN) or a Variational Autoencoder (VAE). Below is an example of a simple GAN implementation in Python using TensorFlow and Keras:

Prerequisites

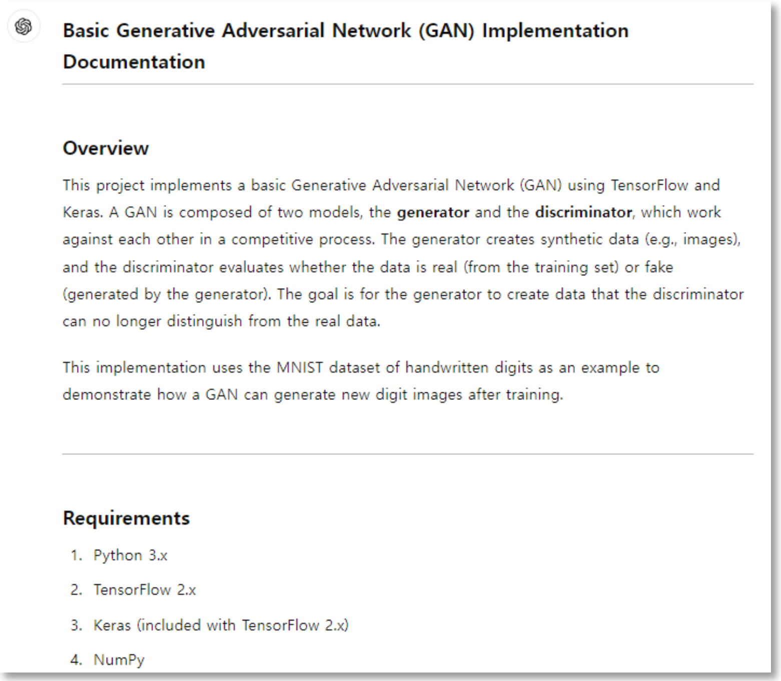
You need to have TensorFlow and other necessary libraries installed:

```bash
pip install tensorflow
```

Code: Basic GAN Example

```python
import tensorflow as tf
from tensorflow.keras import layers
import numpy as np
import matplotlib.pyplot as plt

# Define input shape and latent dimension
img_shape = (28, 28, 1)
latent_dim = 100

# Generator model
def build_generator():
    model = tf.keras.Sequential()
```

Create Documentation



Basic Generative Adversarial Network (GAN) Implementation Documentation

Overview

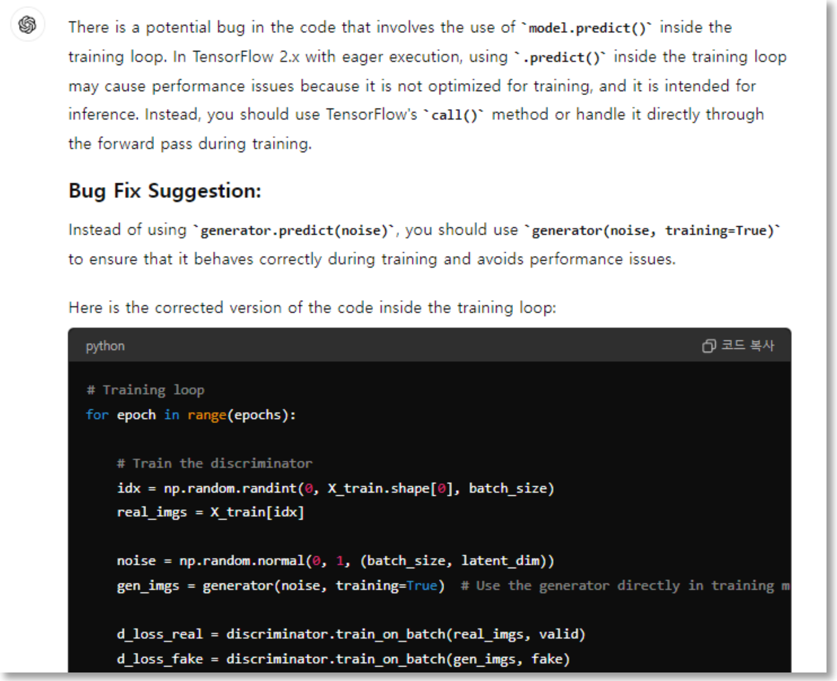
This project implements a basic Generative Adversarial Network (GAN) using TensorFlow and Keras. A GAN is composed of two models, the **generator** and the **discriminator**, which work against each other in a competitive process. The generator creates synthetic data (e.g., images), and the discriminator evaluates whether the data is real (from the training set) or fake (generated by the generator). The goal is for the generator to create data that the discriminator can no longer distinguish from the real data.

This implementation uses the MNIST dataset of handwritten digits as an example to demonstrate how a GAN can generate new digit images after training.

Requirements

1. Python 3.x
2. TensorFlow 2.x
3. Keras (included with TensorFlow 2.x)
4. NumPy

Assist Debugging



There is a potential bug in the code that involves the use of `model.predict()` inside the training loop. In TensorFlow 2.x with eager execution, using `.predict()` inside the training loop may cause performance issues because it is not optimized for training, and it is intended for inference. Instead, you should use TensorFlow's `call()` method or handle it directly through the forward pass during training.

Bug Fix Suggestion:

Instead of using `generator.predict(noise)`, you should use `generator(noise, training=True)` to ensure that it behaves correctly during training and avoids performance issues.

Here is the corrected version of the code inside the training loop:

```python
# Training loop
for epoch in range(epochs):

    # Train the discriminator
    idx = np.random.randint(0, X_train.shape[0], batch_size)
    real_imgs = X_train[idx]

    noise = np.random.normal(0, 1, (batch_size, latent_dim))
    gen_imgs = generator(noise, training=True)  # Use the generator directly in training m

    d_loss_real = discriminator.train_on_batch(real_imgs, valid)
    d_loss_fake = discriminator.train_on_batch(gen_imgs, fake)
```

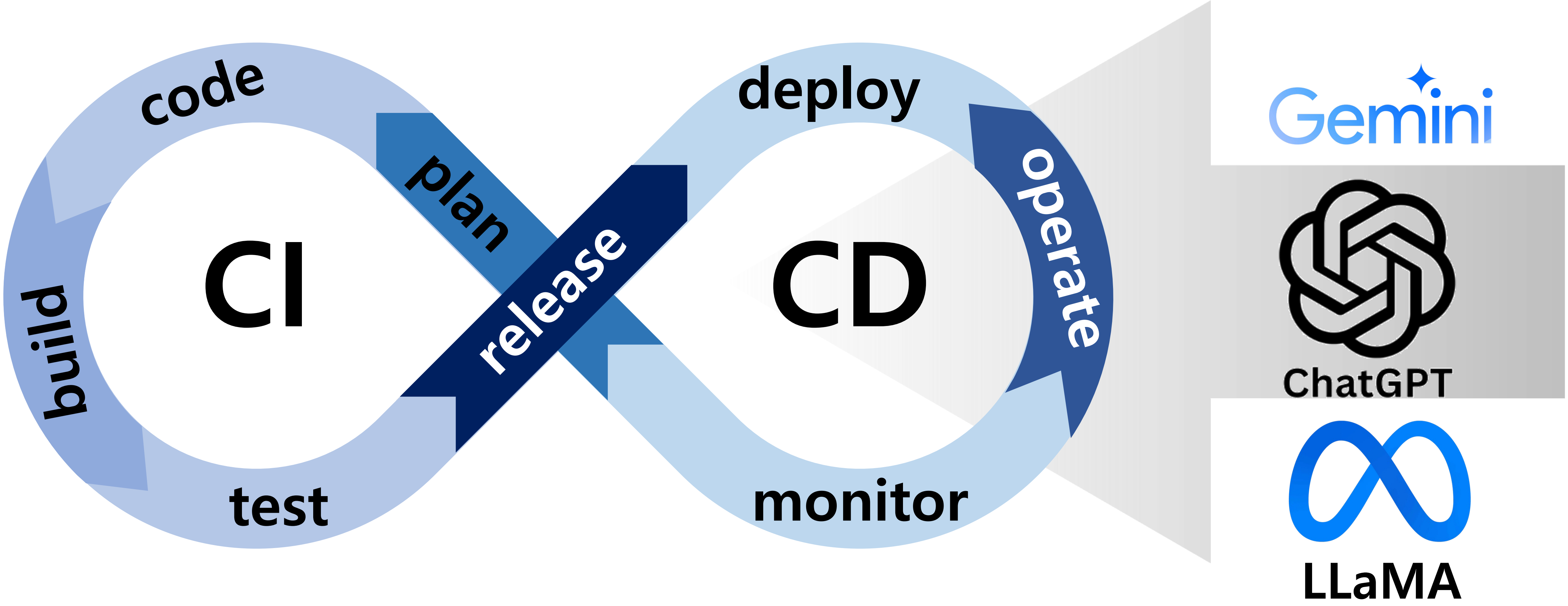
Generative AI in Software Development

- **Main benefits of generative AI**
  - Increased productivity
  - Enhanced creativity
  - Improved code quality

➔ Humans can focus on more valuable tasks

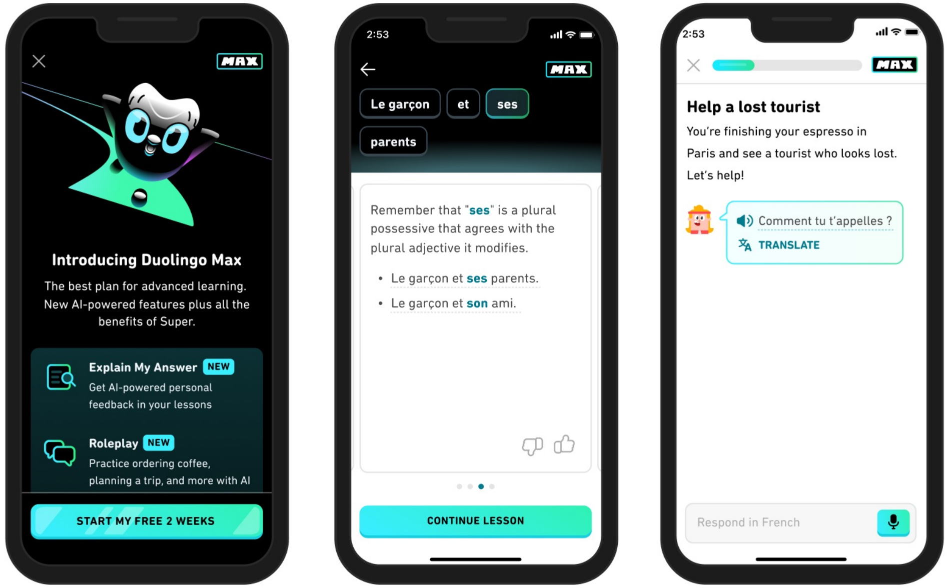
Generative AI in Software Development

- generative AI is being integrated into CI/CD pipelines



AI in Education

- **Generative AI is being utilized in education**


MAX
Introducing Duolingo Max
The best plan for advanced learning.
New AI-powered features plus all the benefits of Super.
Explain My Answer NEW
Get AI-powered personal feedback in your lessons
Roleplay NEW
Practice ordering coffee, planning a trip, and more with AI
START MY FREE 2 WEEKS
2:53
MAX
Le garçon
et
ses
parents
CONTINUE LESSON
2:53
MAX
TRANSLATE
Respond in French

Remember that "ses" is a plural possessive that agrees with the plural adjective it modifies.

- Le garçon et **ses** parents.
- Le garçon et **son** ami.

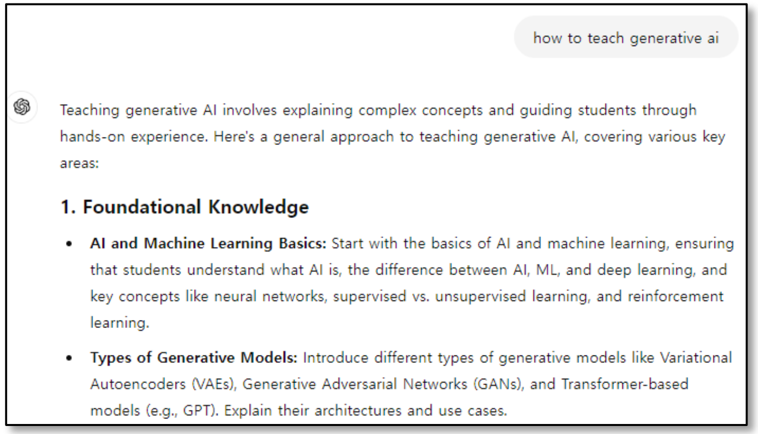
Help a lost tourist

You're finishing your espresso in Paris and see a tourist who looks lost. Let's help!

Comment tu t'appelles ?

Duolingo

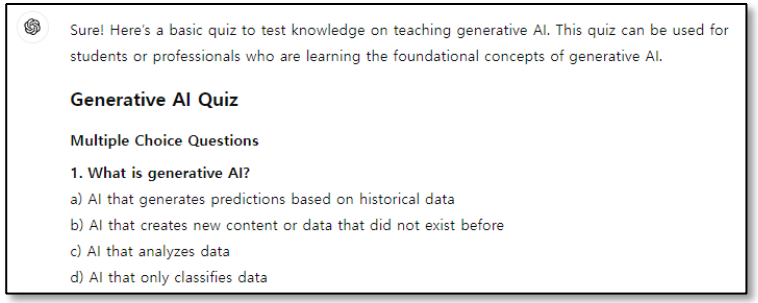
Speak Now



how to teach generative ai

Teaching generative AI involves explaining complex concepts and guiding students through hands-on experience. Here's a general approach to teaching generative AI, covering various key areas:

1. **Foundational Knowledge**
   - **AI and Machine Learning Basics:** Start with the basics of AI and machine learning, ensuring that students understand what AI is, the difference between AI, ML, and deep learning, and key concepts like neural networks, supervised vs. unsupervised learning, and reinforcement learning.
   - **Types of Generative Models:** Introduce different types of generative models like Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs), and Transformer-based models (e.g., GPT). Explain their architectures and use cases.



Sure! Here's a basic quiz to test knowledge on teaching generative AI. This quiz can be used for students or professionals who are learning the foundational concepts of generative AI.

Generative AI Quiz

Multiple Choice Questions

1. What is generative AI?

a) AI that generates predictions based on historical data
b) AI that creates new content or data that did not exist before
c) AI that analyzes data
d) AI that only classifies data

- **Formative Assessment and Real-Time Feedback**
- **Personalized Learning Experiences**

- **Teacher Support**

Software Development Competition Overview

Category	Details
Participants	36 students participated in the competition
Competition Duration	4 weeks
Competition Rounds	2 rounds
Advancing to Round 2	Top 15 students selected from Round 1
Project Requirements	Individual software project with mandatory ChatGPT usage, applied optionally across various stages of the development lifecycle
Evaluation	5 evaluators: 2 external and 3 internal
Objective	Determine the overall winner based on project per formance and utilization of ChatGPT

Competition Structure and Phases

- **Phases of the Competition**
    - Idea Planning and Brainstorming
    - Documentation and Requirement Gathering (Market/Product Requirements)
    - Coding and Implementation
    - Debugging and Error Resolution
    - Quality Assurance and Testing

Pre-Competition Survey Results and Impact Analysis

- **Pre-Competition Survey**

(Unit: %)

Question	Finalist	Non-Finalist	All
GPT Usage Experience	100	82	88
GPT Integration in Development	75	53	60
LLM Learning Experience	25	6	12
High Interest in Software Development	75	18	36

- **Impact Analysis**
  - Positive correlation between ChatGPT usage and final project scores ($r = 0.65$)
  - Statistically significant relationship between ChatGPT usage experience and project scores (p-value < 0.01)

Student Contributions

- **ChatGPT usage across development stages**

(Unit: %)

Category	Usage Rate
Planning (Ideation, MRD/PRD Generation)	100
Design (UI/UX design, Graphic Resources)	66.7
Scenario Creation (Story writing, Scripting)	33.3
Programming (Coding, bug fixing)	80
Server (API integration, Server)	46.7
Security (Data Security, Privacy Protection)	20
Sound (Sound effects and music Generation)	13.3
QA (Quality Assurance)	26.7

***Award-Winning Project**

By JunHyeong Park

Songs of the Seven Dragons

A text-based RPG developed by generating resources using GPT and Stable Diffusion

To what extent can ChatGPT take on the role of a game developer?

Developing a game as a solo developer, with GPT performing as many roles as possible during development.

Ideation

Art Design

Story

Programming

Server

Security

Sound

QA

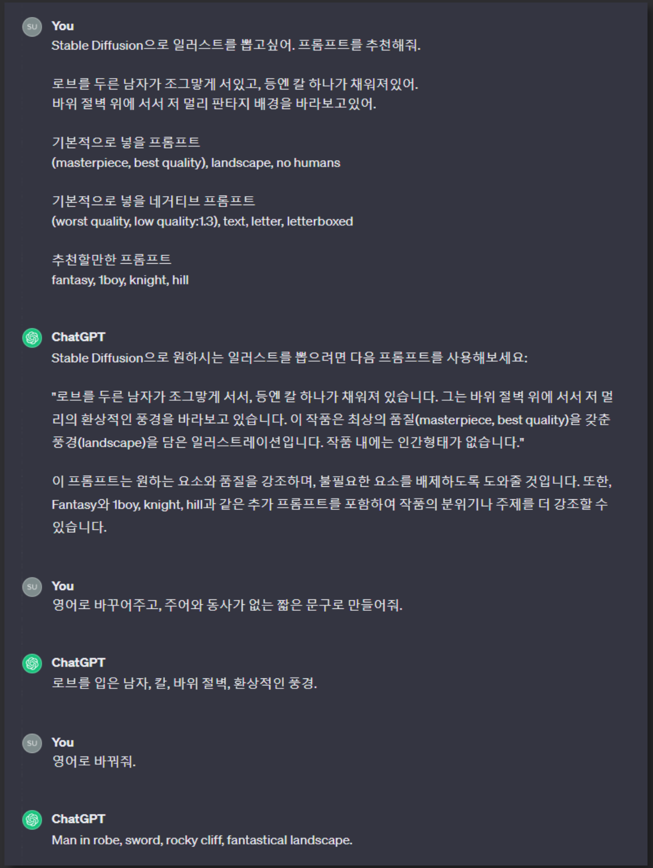
***Award-Winning Project**

Songs of the Seven Dragons

A text-based RPG developed by generating resources using GPT and Stable Diffusion

Art Design

with Stable Diffusion



You
Stable Diffusion으로 일러스트를 뽑고싶어. 프롬프트를 추천해줘.

로브를 두른 남자가 조그맣게 서있고, 등엔 칼 하나가 채워져있어.
바위 절벽 위에 서서 저 멀리 판타지 배경을 바라보고있어.

기본적으로 넣을 프롬프트
(masterpiece, best quality), landscape, no humans

기본적으로 넣을 네거티브 프롬프트
(worst quality, low quality:1.3), text, letter, letterboxed

추천할만한 프롬프트
fantasy, 1boy, knight, hill

ChatGPT
Stable Diffusion으로 원하시는 일러스트를 뽑으려면 다음 프롬프트를 사용해보세요:

"로브를 두른 남자가 조그맣게 서서, 등엔 칼 하나가 채워져 있습니다. 그는 바위 절벽 위에 서서 저 멀리의 환상적인 풍경을 바라보고 있습니다. 이 작품은 최상의 품질(masterpiece, best quality)을 갖춘 풍경(landscape)을 담은 일러스트레이션입니다. 작품 내에는 인간형태가 없습니다."

이 프롬프트는 원하는 요소와 품질을 강조하며, 불필요한 요소를 배제하도록 도와줄 것입니다. 또한, Fantasy와 1boy, knight, hill과 같은 추가 프롬프트를 포함하여 작품의 분위기나 주제를 더 강조할 수 있습니다.

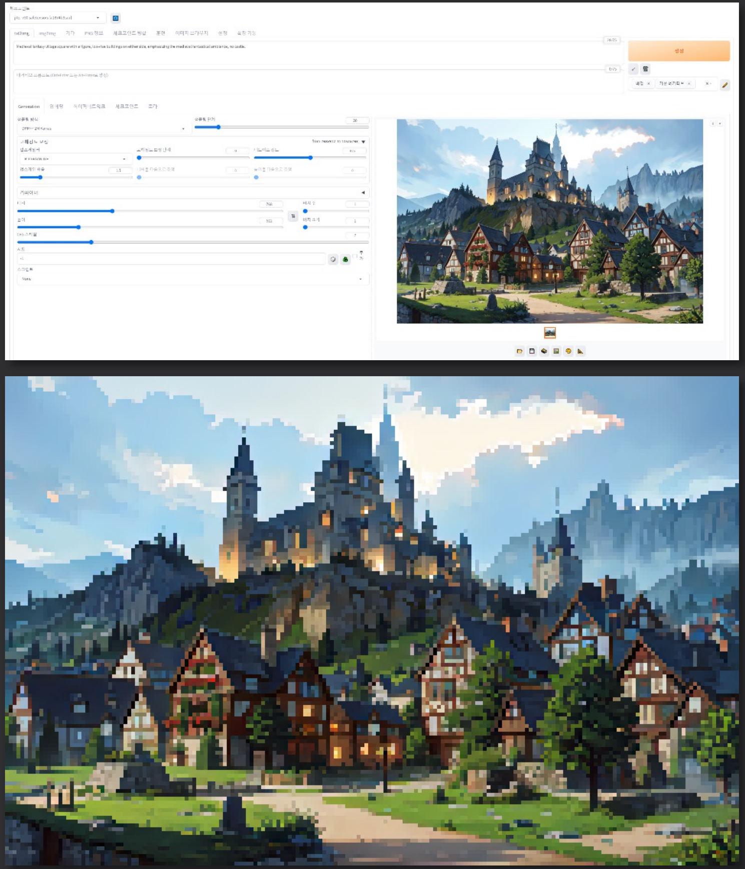
You
영어로 바꾸어주고, 주어와 동사가 없는 짧은 문구로 만들어줘.

ChatGPT
로브를 입은 남자, 칼, 바위 절벽, 환상적인 풍경.

You
영어로 바꿔줘.

ChatGPT
Man in robe, sword, rocky cliff, fantastical landscape.

Prompt Generation by ChatGPT for Graphic Resource Generation



Generated Images from the prompt generated by ChatGPT

*Award-Winning Project

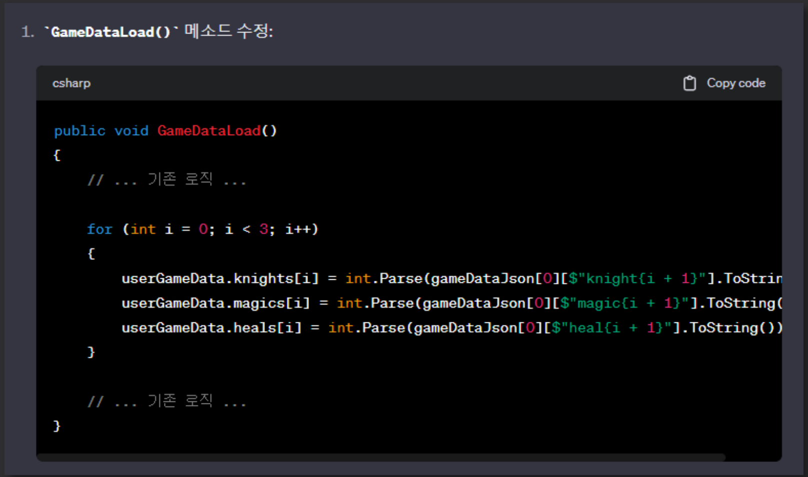
By JunHyeong Park

Songs of the Seven Dragons

A text-based RPG developed by generating resources using GPT and Stable Diffusion

Server

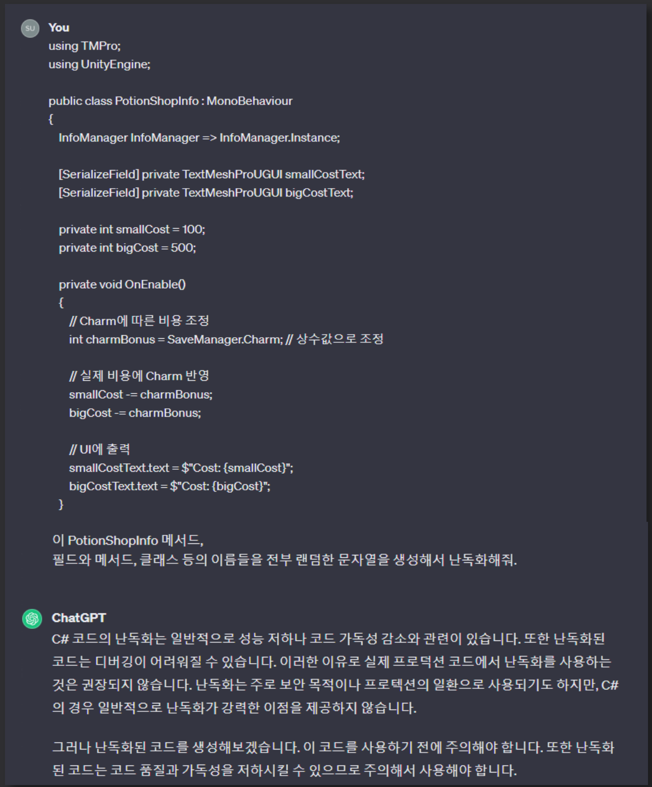
Security



1. `GameDataLoad()` 메소드 수정:

```csharp
public void GameDataLoad()
{
    // ... 기존 로직 ...

    for (int i = 0; i < 3; i++)
    {
        userGameData.knights[i] = int.Parse(gameDataJson[0][$"knight{i + 1}"].ToStrin
        userGameData.magics[i] = int.Parse(gameDataJson[0][$"magic{i + 1}"].ToString(
        userGameData.heals[i] = int.Parse(gameDataJson[0][$"heal{i + 1}"].ToString())
    }

    // ... 기존 로직 ...
}
```

Implementing methods for exchanging data with a game backend server in JSON format



You

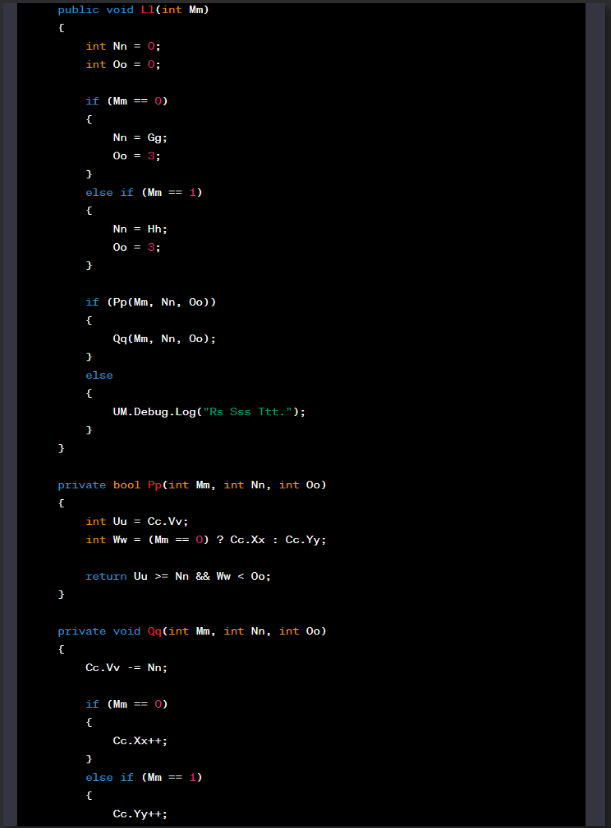
```
using TMPro;
using UnityEngine;

public class PotionShopInfo : MonoBehaviour
{
  InfoManager InfoManager => InfoManager.Instance;

  [SerializeField] private TextMeshProUGUI smallCostText;
  [SerializeField] private TextMeshProUGUI bigCostText;

  private int smallCost = 100;
  private int bigCost = 500;

  private void OnEnable()
  {
    // Charm에 따른 비용 조정
    int charmBonus = SaveManager.Charm; // 상수값으로 조정

    // 실제 비용에 Charm 반영
    smallCost -= charmBonus;
    bigCost -= charmBonus;

    // UI에 출력
    smallCostText.text = $"Cost: {smallCost}";
    bigCostText.text = $"Cost: {bigCost}";
  }
```

이 PotionShopInfo 메서드,
필드와 메서드, 클래스 등의 이름들을 전부 랜덤한 문자열을 생성해서 난독화해줘.

ChatGPT

C# 코드의 난독화는 일반적으로 성능 저하나 코드 가독성 감소와 관련이 있습니다. 또한 난독화된 코드는 디버깅이 어려워질 수 있습니다. 이러한 이유로 실제 프로덕션 코드에서 난독화를 사용하는 것은 권장되지 않습니다. 난독화는 주로 보안 목적이나 프로텍션의 일환으로 사용되기도 하지만, C#의 경우 일반적으로 난독화가 강력한 이점을 제공하지 않습니다.

그러나 난독화된 코드를 생성해보겠습니다. 이 코드를 사용하기 전에 주의해야 합니다. 또한 난독화된 코드는 코드 품질과 가독성을 저하시킬 수 있으므로 주의해서 사용해야 합니다.



```csharp
public void Ll(int Mm)
{
    int Nn = 0;
    int Oo = 0;

    if (Mm == 0)
    {
        Nn = Gg;
        Oo = 3;
    }
    else if (Mm == 1)
    {
        Nn = Hh;
        Oo = 3;
    }

    if (Pp(Mm, Nn, Oo))
    {
        Qq(Mm, Nn, Oo);
    }
    else
    {
        UM.Debug.Log("Rs Sss Ttt.");
    }
}

private bool Pp(int Mm, int Nn, int Oo)
{
    int Uu = Cc.Vv;
    int Ww = (Mm == 0) ? Cc.Xx : Cc.Yy;

    return Uu >= Nn && Ww < Oo;
}

private void Qq(int Mm, int Nn, int Oo)
{
    Cc.Vv -= Nn;

    if (Mm == 0)
    {
        Cc.Xx++;
    }
    else if (Mm == 1)
    {
        Cc.Yy++;
```

Obfuscating code by generating random strings for fields, methods, properties, parameters, and class names in the desired script by ChatGPT

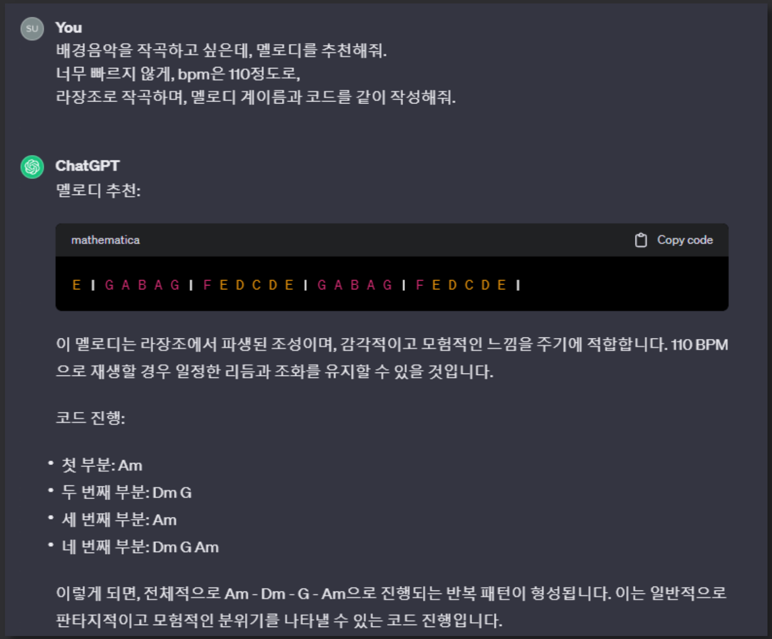
***Award-Winning Project**

By JunHyeong Park

Songs of the Seven Dragons

A text-based RPG developed by generating resources using GPT and Stable Diffusion

Sound



You
배경음악을 작곡하고 싶은데, 멜로디를 추천해줘.
너무 빠르지 않게, bpm은 110정도로,
라장조로 작곡하며, 멜로디 계이름과 코드를 같이 작성해줘.

ChatGPT
멜로디 추천:

```mathematica
E | G A B A G | F E D C D E | G A B A G | F E D C D E |
```

이 멜로디는 라장조에서 파생된 조성이며, 감각적이고 모험적인 느낌을 주기에 적합합니다. 110 BPM으로 재생할 경우 일정한 리듬과 조화를 유지할 수 있을 것입니다.

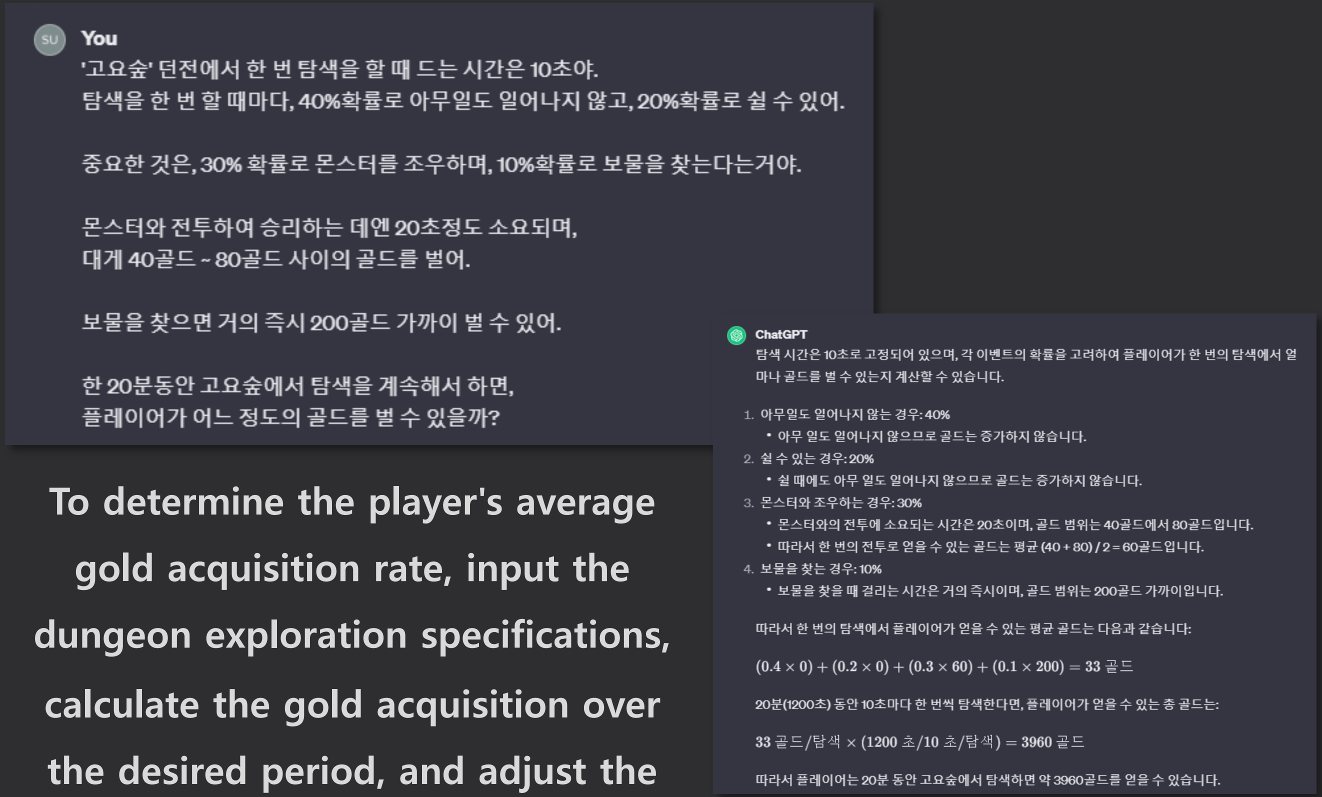
코드 진행:

- 첫 부분: Am
- 두 번째 부분: Dm G
- 세 번째 부분: Am
- 네 번째 부분: Dm G Am

이렇게 되면, 전체적으로 Am - Dm - G - Am으로 진행되는 반복 패턴이 형성됩니다. 이는 일반적으로 판타지적이고 모험적인 분위기를 나타낼 수 있는 코드 진행입니다.

Melody suggestions and music code generation

QA



You
'고요숲' 던전에서 한 번 탐색을 할 때 드는 시간은 10초야.
탐색을 한 번 할 때마다, 40%확률로 아무일도 일어나지 않고, 20%확률로 쉴 수 있어.

중요한 것은, 30% 확률로 몬스터를 조우하며, 10%확률로 보물을 찾는다는거야.

몬스터와 전투하여 승리하는 데엔 20초정도 소요되며,
대게 40골드 ~ 80골드 사이의 골드를 벌어.

보물을 찾으면 거의 즉시 200골드 가까이 벌 수 있어.

한 20분동안 고요숲에서 탐색을 계속해서 하면,
플레이어가 어느 정도의 골드를 벌 수 있을까?

ChatGPT
탐색 시간은 10초로 고정되어 있으며, 각 이벤트의 확률을 고려하여 플레이어가 한 번의 탐색에서 얼마나 골드를 벌 수 있는지 계산할 수 있습니다.

1. 아무일도 일어나지 않는 경우: 40%
   - 아무 일도 일어나지 않으므로 골드는 증가하지 않습니다.
2. 쉴 수 있는 경우: 20%
   - 쉴 때에도 아무 일도 일어나지 않으므로 골드는 증가하지 않습니다.
3. 몬스터와 조우하는 경우: 30%
   - 몬스터와의 전투에 소요되는 시간은 20초이며, 골드 범위는 40골드에서 80골드입니다.
   - 따라서 한 번의 전투로 얻을 수 있는 골드는 평균 (40 + 80) / 2 = 60골드입니다.
4. 보물을 찾는 경우: 10%
   - 보물을 찾을 때 걸리는 시간은 거의 즉시이며, 골드 범위는 200골드 가까이입니다.

따라서 한 번의 탐색에서 플레이어가 얻을 수 있는 평균 골드는 다음과 같습니다:

$$(0.4 \times 0) + (0.2 \times 0) + (0.3 \times 60) + (0.1 \times 200) = 33 \text{ 골드}$$

20분(1200초) 동안 10초마다 한 번씩 탐색한다면, 플레이어가 얻을 수 있는 총 골드는:

$$33 \text{ 골드/탐색} \times (1200 \text{ 초}/10 \text{ 초/탐색}) = 3960 \text{ 골드}$$

따라서 플레이어는 20분 동안 고요숲에서 탐색하면 약 3960골드를 얻을 수 있습니다.

To determine the player's average gold acquisition rate, input the dungeon exploration specifications, calculate the gold acquisition over the desired period, and adjust the balance based on the results.

By JunHyeong Park

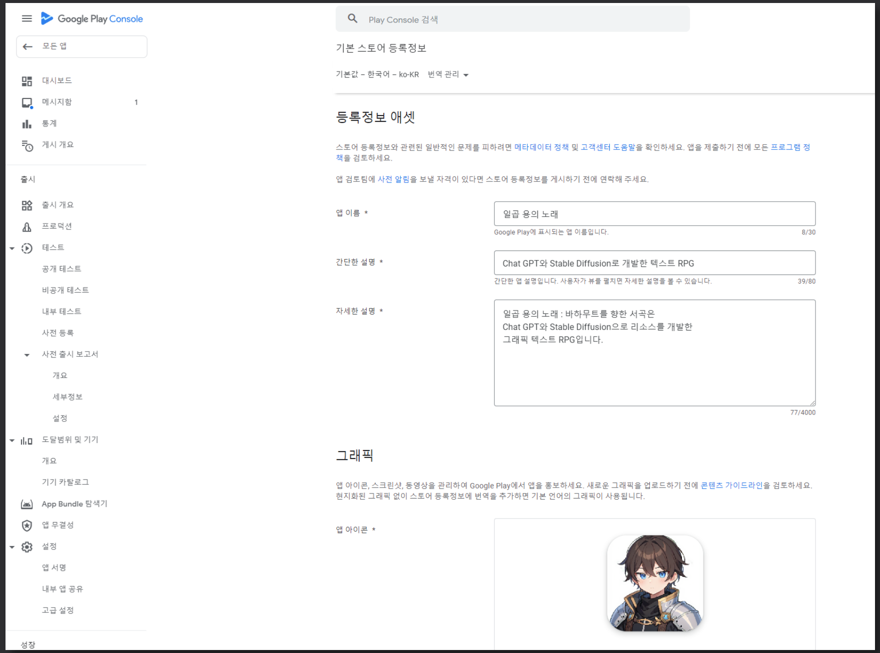
*Award-Winning Project

Songs of the Seven Dragons

A text-based RPG developed by generating resources using GPT and Stable Diffusion

Open Source

https://github.com/Suldangoo/A-Songs-of-the-Seven-Dragons



기본 스토어 등록정보

기본값 – 한국어 – ko-KR 번역 관리

등록정보 애셋

스토어 등록정보와 관련된 일반적인 문제를 피하려면 메타데이터 정책 및 고객센터 도움말을 확인하세요. 앱을 제출하기 전에 모든 프로그램 정책을 검토하세요.

앱 검토팀에 사전 알림을 보낼 자격이 있다면 스토어 등록정보를 게시하기 전에 연락해 주세요.

앱 이름 *
일곱 용의 노래
Google Play에 표시되는 앱 이름입니다. 8/30

간단한 설명 *
Chat GPT와 Stable Diffusion로 개발한 텍스트 RPG
간단한 앱 설명입니다. 사용자가 뷰를 펼치면 자세한 설명을 볼 수 있습니다. 39/80

자세한 설명 *
일곱 용의 노래 : 바하무트를 향한 서곡은
Chat GPT와 Stable Diffusion으로 리소스를 개발한
그래픽 텍스트 RPG입니다.
77/4000

그래픽

앱 아이콘, 스크린샷, 동영상을 관리하여 Google Play에서 앱을 홍보하세요. 새로운 그래픽을 업로드하기 전에 콘텐츠 가이드라인을 검토하세요.
현지화된 그래픽 없이 스토어 등록정보에 번역을 추가하면 기본 언어의 그래픽이 사용됩니다.

앱 아이콘 *

Demo

Insights, Challenges, and Limitations

- ChatGPT integration improved project outcomes, particularly in planning, coding, and quality assurance, with higher completion rates and better scores.
- Lower adoption in specialized areas (server setup, security, sound, QA) indicates limitations in its current capabilities for technical tasks.
- Study limitations include a small sample size and a focus on a specific competition, highlighting the need for broader research in varied educational contexts.

Future Work

- **Expand research to include more diverse educational settings and larger participant groups.**
    - Around 90 students (Fall 2024, Course Title: Generative AI and its Applications)

- **Refine AI tools like ChatGPT to better support specialized tasks in software development.**
    - Collaborate with domain experts to develop strategies.

- **Explore the long-term impacts of AI on student learning and career development to improve AI-driven education.**
    - Observe changes in students as they learn and apply generative AI over time.

Thank You

Feel free to contact

E-mail	sunheehwang@dongyang.ac.kr
Linkedin	https://www.linkedin.com/in/sun-hee-hwang/

Sunhee Hwang
Assistant Professor at Dongyang
Mirae University