# CIS 700 Final Project Proposal

## Project Description

Nobody wants to write docstrings. What if we can build a better description generator? Recall the games you played; the item descriptions have writing styles particular to those games. For example, you know the item is from Dark Soul or Don't Starve simply by reading the text: even when they are not! Therefore, our work is more developer-oriented, particularly convenient for those who want to develop mods for their favorite games and conform to a consistent writing style.

So, in this project, we would like to revisit homework 2, the description generation. We want a generator that takes an item title as input and generates a corresponding gamified item description. However, instead of building an end-to-end system as we have done in homework 2, we observe that such a task is two-fold:

1. Generate a basic description for a simplified item name that depicts the main properties of this item. For instance, a sword is made of metal, a pointy weapon.
2. Render the utterance to sound like a game item when we condition it on the game category and full item name like King Arthur’s Sword.

To sum, we want to formulate this task as a combination of description generation and conditioned style transfer problem.

## Proposed Methods

As discussed above, we will build two sub-models to fulfill our objective. The first model that generates descriptions given an item name should be a large Pre-trained Language Model (PLM) equipped with common sense knowledge, either implicitly learned (GPT-3) or explicitly learned (COMET). We have not yet decided whether tune this model or not (we will start with prompting GPT-3).

Next, we will fine-tune a renderer, which is another Seq2Seq PLM model, with game-specific data to perform style transfer. Our proposed method is first to extract the keyword from the game item, e.g., pull "Sword" from "King Arthur’s Sword," and feed this primary item name to the pre-trained generator to get a base description. After that, we can derive a standard style-transfer dataset for sequence-to-sequence fine-tuning.

## Data

Suppose we are going to fine-tune the PLM generator. In that case, we consider using a proxy dataset, the [Amazon Review Data](https://nijianmo.github.io/amazon/index.html#sample-metadata), which has a comprehensive collection of products with both item names and short descriptions. As for the renderer, we assume the [LIGHT](https://arxiv.org/pdf/1903.03094.pdf) dataset is sufficient. However, if time permits, we also want to scrape item description data from [Fandom Game Wiki](https://www.fandom.com/topics/games), e.g., for [Terraria](https://terraria.fandom.com/wiki/Terraria_Wiki) items. Thus, we can significantly augment our dataset, but we need to manually determine the (LIGHT) game category.

## Related Works

To the best of our knowledge, no previous work applies description generation and style-transfer in this particular fashion. But we found related work about generating [personalized product descriptions for e-commerce](https://dl.acm.org/doi/pdf/10.1145/3292500.3330725). And this paper inspired us to construct gamified item description generator.

## Team Members

* Ning Wan: handling Amazon Review Data and experiment with generator
* Hongyu Zhang: handling LIGHT data and experiment with style-transfer
* Yuxuan Wang: same as above