Analyzing NPC Relationships Using Graph in Stardew Valley

Muhammad Rafi Akbar - 13524125
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail: raf14kbar9a@gmail.com , 13524125@std.stei.itb.ac.id

***Abstract*—*Stardew Valley* is a farming simulation game where the player can interact with non-playable characters (NPCs) that plays a crucial role in gameplay. These interactions, which include gift-giving, conversations, and events, form a relationship between the player and NPCs. Each NPC also has a relationship with each other, with some being married, friends, or siblings. There are two objectives of this essay, the first is to visualize the relationship between NPCs using graph and to determine the gifts that can be given to NPCs in order to create an optimal way to increase their relationship level using graphs. By representing NPCs, and Items as vertices and edges, we can analyze the structure of social relationships to gain insights into NPC behavior and player strategy.**

***Keywords*— NPCs, Items, relationship, graph,**

I. Introduction

Stardew Valley is a farm life simulation indie game developed by Eric Barone in 2016. The player of the game take the role of an unnamed character who inherits a farm from their deceased grandfather in a place called "Stardew Valley". The world of Stardew Valley revolves around four seasons that all last 28 days, which is Spring, Summer, Fall, and Winter. The townspeople will occasionally celebrate events during the seasons, which can give you rewards if you decide to participate in it. In this game, the players are given tools that can be used to interact with the environment which will reduce the player's stamina over time. players can manage a farm by clearing land, growing crops, and raising animals. Players can also take part in other activities such as fishing, mining, or even engage in combat against monsters in an endless dungeon-like environment. Each activity the player do can increase the level of skills the player has, which can give them buffs such as better quality of crops, reduced energy cost for tools, or allow them to gain access to better equipment. Players can also improve themselves by equipping accessories and items. The players can also interact with the NPCs in the game, raising their relationship levels by constantly engaging in a conversation with them and giving them gifts that they favor, which can significantly increase the points in their relationship level. Once the relationship level is high enough, the player can even choose the option to marry an NPC.

Figure 1. Stardew Valley official Steam poster. Source: https://store.steampowered.com/app/413150/Stardew_Valley/

The main objective in Stardew Valley is to restore the farm that was inherited by the player from their grandfather, and also help out the town by doing tasks that were given to us by the NPCs. After exploring the north side of town, the player will find a broken building which used to be the town's Community Center. The player will be given the task of restoring the Community Center by completing the quest available inside the Community Center in the form of "*bundles of goods*". As the player progresses the game and finishes all the quests in the Community Center, the game will unlock new areas and activities.

In order to restore the farm, the players will need to earn money to buy farm equipment and tools. Players can earn money by doing activities such as growing a plant, mining, slaying monsters, and fishing which can all be sold to different NPCs for money. Players can grow a plant by planting seeds into the ground and watering them until they fully grow. The player can gain seeds by exploring or buying them from NPCs. Players can also upgrade their gears and equipment by bringing specific materials they obtained from mining and talking to an NPC, which will take the equipment you want to upgrade out of your inventory for a few days until he finishes upgrading it.

Based of what we know from the game so far, we can conclude that NPCs are one of the most significant aspects of the game because they are needed in order for the player to progress the game further. This paper aims to analyze the relationship between NPCs and the ideal items the player can give to NPCs to quickly increase their relationship level.

II. THEORETICAL FRAMEWORK

A. *Graph Theory*

Graph theory is a method to represent the relationship between a discrete object. A graph consists of vertices with edges that connect the vertices, used to represent the relationship between the vertices.

A common notation for graph is:

$$G = (V, E)$$

$V = \{v1, v2, v3, \ldots, vn\}$

$E = \{e1, e2, e3, \ldots, en\}$

With V being a non-empty set of vertices

And E being a non empty set of edges

Type of Graphs:

a) *Simple graph*

A graph that does not contain any loops or parallel edges

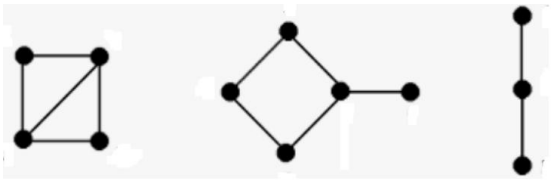

Figure 2. Example of simple graphs. Source: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

b) Non-simple graph

A graph that contain loops or parallel edges

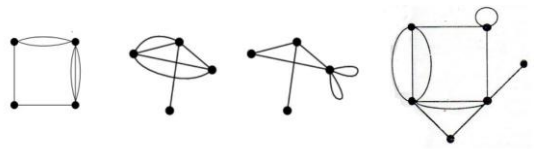

Figure 3. Example of non-simple graphs. Source: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

Based on the orientation of the direction on the side, graphs are divided into 2 types:

a) Undirected graph

A graph in which the edges have no orientation

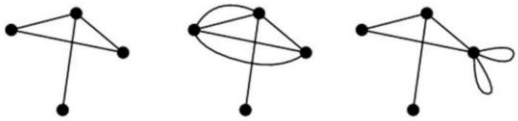

Figure 4. Example of undirected Graph. Source: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

b) Directed graph (digraph)

A graph in which each edges is given an orientated direction

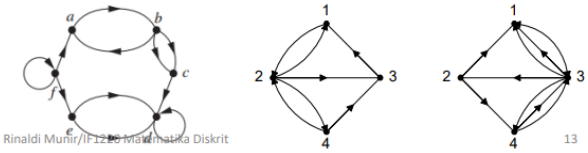

Figure 5. Example of a directed graph. Source: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

Graph Terminology:

1. Adjacent
   A relationship between two vertices that is created when both of the vertices are connected directly with an edge.
2. Incidency
   A relationship between an edge and the two vertices that is connected to the edge.
3. Isolated Vertex
   An isolated vertex refers to a vertex that has no edge that is in Incidency with the vertex.
4. Null graph (Empty graph)
   A Null graph refers to a graph in which the vertices have no edges (empty).
5. Degree
   The degree of a vertex is based off the amount of edges that is in incidency with the vertex.
6. Path
   A sequence of edges that connects a series of vertices in a graph from a vertex acting as the starting point to its destination which cannot be the same as the starting point. The length of a path is the amount of edges in the path.
7. Cycle/Circuit
   A Path that starts and ends in the same vertex
8. Connected
   Two vertices A and B are considered connected if there exist a circuit that starts from A to B.
9. Subgraph
   A smaller part of a larger graph.
   If a Subgraph contain all vertices from a graph, it is called a *Spanning Subgraph.*
10. Cut-Set
    A collection of edges, in which if erased, will cause the graph to become unconnected.
11. Weighted Graph

A graph where each edges have a value that is assigned into it. More than one edges can have the same value.

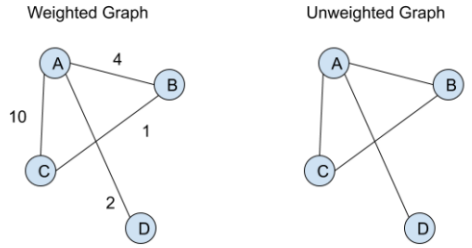

Figure 6. Difference between a weighted graph and an unweighted graph. Source: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

B. Special Graph

a) Complete graph

A simple graph in which all vertices have edges that connect all of them to each other.

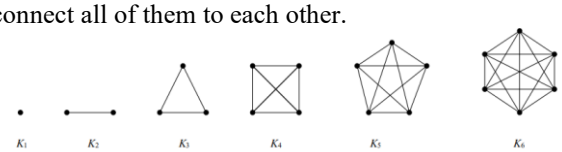

Figure 7. Example of a complete graph. Source: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

b) Circle graph

A graph where all vertices have exactly two degrees

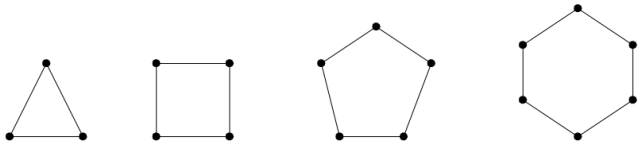

Figure 8. Example of a circle graph. Source: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

c) Regular graph

A graph where all vertices have an equal amount of degrees

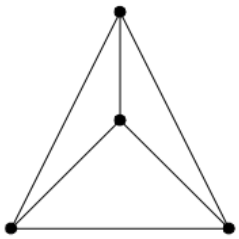

Figure 9. Example of a regular graph. Source: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

d) Bipartite graph

A graph with a set of vertices that could be divided into two collection, and every edge connects a vertex from one group to a vertex in the other group.

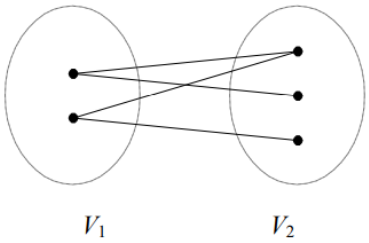

Figure 10. An example of a bipartite graph. Source: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

C. Labeled Graph

A labeled graph is a type of graph which contain additional information attached to its vertex, edges, or even both. The reason we need to use a labeled graph is in order to differentiate the type of relationship the NPCs have with each other (friends, siblings, married, parents, etc).

D. Entity and Relationship

In Stardew Valley, entities in-game can interact with each other, forming a dynamic and interconnected system. The entities include:

1. The player
2. NPCs (Non Playable Characters), with each of them having a unique personality, traits, and schedule
3. Items, which include tools, crops, stones, fish, food, or anything you can obtain and pick up
4. Location, such as the town, shops, mines, and farm

These entities are connected through various type of relationship, each having their own unique function in the game, such as:

1. Social Relationship (friends, family ties, romantic connection between the characters)
2. Gift Preference (each NPC have their own items that they prefer and they dislike, which can affect the relationship score between the players and NPC)
3. Gameplay Dependant (a relationship that occurs in order for players to obtain or receive something from other entities)

By identifying the type of entities and how they relate to one another, which in this case, is between NPCs, and Items and NPCs, we can model these system in the form of graph theorem, represented by vertices and edges.

III. Implementation Method

A. *Establishing the List of NPCs*

Figure 11. List of all NPCs in Stardew Valley.
Source: https://www.reddit.com/r/StardewValley/comments/y20f66/hello_stardew_valley_players_out_here_i_would_be/

There are a total of 32 interactable NPCs in *Stardew Valley*, each of them having their own unique schedule, hobbies, preferences, and relationships. In order to model these relationships using the graph theory, we will compile the list of all NPCs as the vertex set in our graphs.

B. *Building the NPC Social Graph*

By compiling the list of all NPCs, we are able to create a labeled graph

$$G = (V, E)$$

Where each edge e ∈ E (or vertex $v \in V$) has an associated label that represent the relationship between the NPCs.

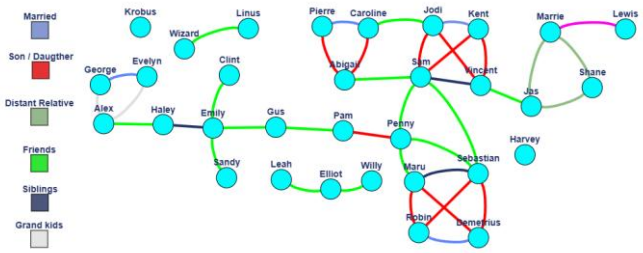

Figure 12. Labeled graph of NPCs relationship

Based on the graph, we can conclude the relationship of all the NPCs. The following table is presented to reference the relationship of all NPCs.

Table 1. List of all NPCs relationship.

NAME	RELATIONSHIP
Pierre	Married to Caroline, Father of Abigail
Caroline	Married to Pierre, Mother of Abigail
Jodi	Married to Kent, Mother of Sam and Vincent, friends with Caroline
Kent	Married to Jodi, Father of Sam and Vincent
Demetrius	Married to Robin, Father of Sebastian and Maru
Robin	Married to Demetrius, Mother of Sebastian and Maru
George	Married to Evelyn, grandfather of Alex
Evelyn	Married to George, grandmother of Alex
Alex	Friends with Haley
Haley	Friends with Alex, sister of Emily
Emily	Sister of Haley, friends with Clint and Sandy
Gus	Friends with Emily and Pam
Pam	Mother of Penny, friends with Gus
Penny	Daughter of Pam, friends with Sam, Sebastian, and Maru
Sam	Son of Jodi and Kent, brother of Vincent, friends with Abigail
Vincent	Son of Jodi and Kent, brother of Sam, friends with Jas
Jas	Niece of Marie, Shane's god daughter
Shane	Nephew of Marie, Jas Godparent
Marnie	Aunt of Jas and Shane, married (dating) with Lewis
Lewis	Married (dating) with Marie
Sebastian	Son of Robin and Demetrius, brother of Maru, friends with Sam and Penny
Maru	Son of Robin and Ddemetrius, sister of Sebastian, friends with Penny
Harvey	-
Leah	Friends with Elliot
Elliot	Friends with Leah and Willy
Willy	Friends with Elliot
Linus	Friends with Wizard
Wizard	Friends with Linus
Clint	Friends with Emily
Sandy	Friends with Emily

Krobus	-
Abigail	Daugther of Pierre and Caroline, friends with Sam

After filtering all NPCs who is married, old and young, we can create a list of NPCs who can become the romantic partner of the player, which include Alex, Emily, Abigail, Sam, Shane, Penny, Sebastian, Maru, and Harvey.

C. Modeling NPC Gift Preference Using Graph

This implementation of this Graph uses the networkx library in Python as the underlying data structure to represent gift preferences in Stardew Valley. The graph is undirected and weighted, where each node represents an NPC or an item, and each edge connects an NPC to an item they like or love, weighted based on the level of their preference.

Loved items have a weight of 80,
Liked items 45,
Neutral items 20,
Dislike items -20,
Hates items -40.

Due to the amount of items that exist in Stardew Valley, we decided to make the program so that it will only show 3 items from each category, and delete the dislike and hates category, in order for the graph to become visible and comprehensive. The deletion of the dislike and hates category also aligns with the purpose of this essay, as we are searching for their preference.

The data used to build this graph is obtained from the official Stardew Valley Wiki using requests, a build in library in Python used to request access to a website, and BeautifulSoup, a built in Python library used to read and extract from the website.

```python
import requests
from bs4 import BeautifulSoup
import networkx as nx
import matplotlib.pyplot as plt
import re

url = "https://stardewvalleywiki.com/List_of_All_Gifts"
res = requests.get(url)
soup = BeautifulSoup(res.text, "html.parser")

table = soup.find("table", {"class": "sortable"})
rows = table.find_all("tr")[1:]

npc_gifts_filtered = {}
```

Figure 13. Stardew Valley item graph code.

The items listed under each NPCs Loved, Liked, and Neutral columns are filtered to remove irrelevant notes (such as “except Red”), in order to ensure no false item is visualized into the graph.

```python
def parse_items_from_cell(cell_tag):
    text_content = cell_tag.get_text(separator=' --ITEM_SEP-- ', strip=True)
    items_raw = re.split(r',|;|--ITEM_SEP--|\n|&', text_content)

    cleaned_items = []
    irrelevant_keywords = {"red", "green", "blue", "orange", "purple", "yellow", "gray"}

    for item in items_raw:
        item = item.strip()
        if "except" in item.lower():
            continue
        item = re.sub(r"\([^)]*\)", "", item).strip()
        if not item or item in {"(", ")", "-", "–"}:
            continue
        if item.lower() in irrelevant_keywords:
            continue

        cleaned_items.append(item)

    return cleaned_items
```

Figure 14. Stardew Valley item graph code.

After accessing the website and parsing its gift table, the code will loop through each row in the gift table. Rows with fewer than 7 elements are skipped to avoid parsing incomplete data. From the valid rows, the name of the NPC is extracted from the first column. The program then proceed to take 3 items under the loved, liked, and neutral categories, with its weight, saved into a variable named npc_gifts_filtered. The program then proceeds to ask user input of NPC name.

```python
for row in rows:
    cols = row.find_all("td")
    if len(cols) < 7:
        continue
    npc = cols[0].get_text(strip=True)

    loved = parse_items_from_cell(cols[2])
    liked = parse_items_from_cell(cols[3])
    neutral = parse_items_from_cell(cols[4])

    npc_gifts_filtered[npc] = []

    def add_filtered_items(items_list, weight, limit=3):
        count = 0
        for item in items_list:
            if count >= limit:
                break
            if item:
                npc_gifts_filtered[npc].append((item, weight))
                count += 1

    add_filtered_items(loved, 80)
    add_filtered_items(liked, 45)
    add_filtered_items(neutral, 20)

target_npc = input("Masukkan nama NPC yang ingin dilihat: ").strip()
```

Figure 15. Stardew Valley item graph code.

The program will then check the validity of the user’s input

```python

if target_npc not in npc_gifts_filtered:
    print(f"NPC '{target_npc}' tidak ditemukan atau tidak punya preferensi yang relevan.")
else:
    subG = nx.Graph()
    subG.add_node(target_npc)
    items_for_display = npc_gifts_filtered[target_npc]

    if not items_for_display:
        print(f"NPC '{target_npc}' tidak memiliki item relevan.")
    else:
        for item_name, weight in items_for_display:
            subG.add_node(item_name)
            subG.add_edge(item_name, target_npc, weight=weight)

```

Figure 16. Stardew Valley item graph code.

Finally, the graph is rendered using matplotlib, with layout of the NPC node on the right side and all item nodes vertically aligned on the left. This makes it easy to understand which gifts an NPC prefers at a glance.

```
# === Visualisasi ===
pos = {target_npc: (1, 0.5)}
num_items = len(items_for_display)

vertical_spacing = 0.9
total_height = (num_items - 1) * vertical_spacing if num_items > 1 else 0
start_y = 0.5 - (total_height / 2)

for i, (item_name, _) in enumerate(items_for_display):
    pos[item_name] = (0, start_y + i * vertical_spacing)

plt.figure(figsize=(15, max(9, num_items * 0.9)))
nx.draw_networkx_nodes(subG, pos, nodelist=[target_npc], node_color='lightblue', node_size=2000)
nx.draw_networkx_nodes(subG, pos, nodelist=[item for item, _ in items_for_display], node_color='lightgreen', node_size=1000)
nx.draw_networkx_edges(subG, pos, width=2)

node_labels = {node: node for node in subG.nodes()}
nx.draw_networkx_labels(subG, pos, labels=node_labels, font_size=7, font_weight='bold')

edge_labels = nx.get_edge_attributes(subG, 'weight')
nx.draw_networkx_edge_labels(
    subG, pos, edge_labels=edge_labels,
    font_size=12,
    font_color='darkred',
    font_weight='bold',
    bbox=dict(facecolor='white', alpha=0.9, edgecolor='black', boxstyle='round,pad=0.3')
)

plt.title(f"Preferensi Hadiah untuk {target_npc} (Loved, Liked, Neutral - maks 3 per kategori)", fontsize=14)
plt.axis("off")
plt.tight_layout()
plt.show()
```

Figure 17. Stardew Valley item graph code.

IV. Result

After Running the program, we are able to visualize the preferred items of each NPC in the form of a weighted graph. Here are the result of Abigail's preferred items from the Loved, Liked, and Neutral category.

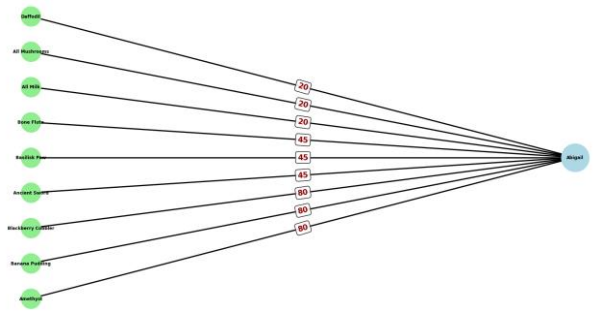

Figure 18. Result of code.

With Abigail's Loved item being Amethyst, Banana Pudding, and Blackberry Cobbler. Liked item being Ancient Sword, Basilisk Paw, and Bone Fulte, and Neutral being All milk, All mushroom, and Daffodil.

V. Conclusion

This essay demonstrate the use of graph theory can be used to visualize and analyze social relationship and gift preferences in the game Stardew Valley. By representing NPC relationships as an undericted label graph, we can observe the social structure of the NPCs living in Stardew Valley, such as family connections, friendships, and romantic connections.

Furthermore, by creating a weighted graph connecting items to NPC based on their gift preferences, we can gain insight into the optimal gift we can give to the NPCs in order to increase the relationship level of the NPC. The weight of the graph directly reference the points we gain if we give the item to the NPC. The graph theory allow us to identify high-utility items and target them toward multiple NPCs in order to gain an optimal progression with their relationship.

VI. Links

Essay explanation video in youtube:

https://youtu.be/ig9ux3rCzSA

GitHub Repository:

https://github.com/Fibarrr/MatematikaDiskrit/tree/main

Acknowledgment

This essay could not have been completed without the help, assistance, and guidance of others. Therefore, the author would like to give his utmost gratitude to:

1. Allah SWT,
2. The lecturer , Dr. Ir. Rinaldi Munir, M.T. in charge of lecturing Discrete Mathematics,
3. The author's friends,
4. The developer of Stardew Valley, Eric Barone,
5. Everyone who contributed to the wiki.

References

[1] https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf accessed on 19th June 2025

[2] https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/21-Graf-Bagian2-2024.pdf accessed on 19th June 2025

[3] https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/22-Graf-Bagian3-2024.pdf accessed on 19th June 2025

[4] https://stardewvalleywiki.com/Stardew_Valley_Wiki accessed on 19th June 2025

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 1 Juni 2025

Muhammad Rafi Akbar (13524125)