Hochschule für Technik
und Wirtschaft Berlin

University of Applied Sciences

VGM
An AI learns to role play

Masterarbeit

Name des Studiengangs
Systemdesign/Gamedesign
FB: 5

vorgelegt von
Benjamin Feder

Datum
Berlin, 30. September 2022

Erstgutachter/in: Prof. Thomas Bremer
Zweitgutachter/in: Prof. Susanne Brandhorst

Abstract

Neural networks have come to a state, where their outputs can become interesting and creative. In this master thesis, the question is asked if and how such neural networks, given their apparent creativity, can be used in context with computer games. Not to steer and control entities like cars or animals, but rather if they can contribute to the actual story content of the game. This premise shall be investigated by developing a game that replicates a pen and paper game. An artificial intelligence capable of generating texts is then implemented to continue and extend the story of this game. This is done by first analyzing and quantifying a pen and paper game. From this abstraction, system elements are extracted, that have to be developed and decided, whether they have to be calculated by common computer algorithms or can be redirected to be evaluated by the artificial intelligence. Finally, the work of the artificial intelligence is being evaluated and compared to the rest of the game. With this evaluation, an answer for the overlying premise is being concluded.

Contents

1 Introduction **1**
1.1 Motivation 1
1.2 Goals 2

I Design **4**

2 Analysis **4**
2.1 Analysis of the System 4
2.2 Analysis of Methods 9
2.3 Analyzing Existing Systems 14

3 System Design **17**
3.1 Agenda 18
3.2 Elements 18
3.3 Interface 19
3.4 Frontend 21

4 Overview of VGM **21**

II Development **24**

5 Content **24**
5.1 Locations 24
5.2 Items 25
5.3 Situations 26
5.4 Events 26

6 Rule Framework **27**
6.1 Objectives 27
6.2 Natural Language Processing 27
6.3 Game State Manager 29

7 AI Framework **30**
7.1 framework Objectives 31
7.2 Development 31
7.3 Implementation 35

III Evaluation **41**

8 AI Evaluation **41**
8.1 Questions 42
8.2 Quantitative Evaluation 45
8.3 Qualitative Evaluation 49
8.4 Results 53

IV Conclusion **59**

9 Further Work **59**
9.1 Scenario Detection 59
9.2 Rephrase Content 59
9.3 NPCs 60

10 Conclusion for VGM **61**

11 Conclusion for AI **62**

12 Attachments **64**
12.1 aidungeon 64
12.2 Risas Secret 73
12.3 Haunted House 75
12.4 Survey 77
12.5 Comments 83

13 Bibliography **92**

Acknowledgements

First, I want to thank everyone who participated in the survey of this master thesis and play tested the framework. Although I would love to thank everyone personally and list all your names, for data privacy I did not gather any of your names. Therefore, I can just call out to all of you: Thanks to all of you, this master thesis has got the finishing touch that seemed to be missing. Your comments and occasional personal conversation afterwards were so inspiring. I would have loved to answer all of your comments, but that would have been way beyond the scope of this thesis. Your responses have shown that all the work that went into VGM resulted in what it was supposed to be.

Special thanks go to Susanne Brandhorst and Thomas Bremer. There was a time during development when I realized, I may have reached out a bit too far. The originally intended scope was way beyond what has been written in this thesis. Your guidance and motivation helped me to restructure the thesis and actually gave it a way more fitting form.

I also want to send thanks to Lukas. Working on a big project like that can become quite frustrating. It really helps to just get your mind off things and share the funny sides of working with AI and neural networks.

Ein ganz besonders großes Dankeschön geht an meine Eltern. Alleine schon dafür, dass ich weit über die Hälfte dieser Arbeit auf ihrer Terasse schreiben konnte und sie das einfach so ertragen haben. Mama hat so oft meinen stundenlangen Erzählungen zugehört, auch wenn sie selber zugab nicht immer alles verstanden zu haben (aber das habe ich ja auch nicht). Und dennoch hat sie immer eine zweite Perspektive gezeigt und alle Themen von einer anderen Seite beleuchtet. Währenddessen hat Papa eine so unglaubliche Menge an IT-Infrastruktur zur Verfügung gestellt, ohne die eine so gezielte und ausgiebige Testphase für das neuronale Netzwerk nicht möglich gewesen wäre. Und wenn mir die Arbeit irgendwann in den Kopf stieg, dann standen immer ein kühles Kellerbier und ein paar Folgen Star Trek zur Verfügung.

But nothing of this thesis would have even been written without the person who actually got me into pen and paper games a few years ago. Yes, Joni-chan, I am talking about you. Thank you so much. Not only

did and do you tell some of the most fascinating stories I had the pleasure to be a part of, you even put up with the shenanigans I brought to the game, no matter how stupid they were. You bring pen and paper from a game, to an experience.

And yes, Wolle, you too. Of course, for all the pen and paper related stuff, I mean it's not like your stories and characters did not have an influence on this thesis. Almost the entire section 2.2 was written during me observing that one game. You know which one. But I also want to thank you for everything else. For being a shoulder to cry on, for always being up for philosophical conversations, for all the distractions you provided when the world was too much to bear (the car just being one of them) and for talking back to me without putting on a mask. There would actually be a lot more to thank you for. But I think this chapter is already getting too long.

My final thanks I will send to Cati for being the most amiable life partner I could wish for. Your patience and motivation was the one constant that kept me going. Having you on my side always made me look forward. Your calm and joy you find even in the simplest of things is so contagious, it always put a smile on my face. Thank you for putting up with me for all these years, especially the last two. If it wasn't for you, so many things would have been very different.

Preface

I know it is not common to start off a master thesis with a personal preamble that talks about something surrounding the topic, instead of the topic itself. However, recent developments and events in the field of artificial intelligence made me see the need to line out my point of view. Using AI is a field filled with controversy, especially when it comes to arts. Since this master thesis involves the art of story writing and telling, during the development of the framework I have had my fair share of conversations about this and other work I did with the use of artificial intelligence.

I can understand the artists concerns, that artificial intelligence may take over their jobs. These pieces of software already deliver artwork that convinces the untrained eye. Just in the few months of developing VGM, the output of some AI has gone from 'hey, that's interesting' to 'how was that written by a computer'. This is especially true for visual arts. I want to point out, that the entirety of this master thesis and the framework that was developed is never ever meant to completely take away from the works of an author or any other artist. Personally, I hold the opinion, that it is not wrong to use AI in arts. However, it is an entirely different topic to separate artist and AI. In my opinion, the problem starts with thinking, that artificial intelligence has to be used on its own. I think that a way more feasible approach is to use AI together with artists, the same way I use a software to check this master thesis for typos, grammar mistakes and maybe suggest better formulations. I do not have the software *write* this thesis, it just *helps* me with it.

This mentality actually helps to see the benefits of using AI. For example, in VGM, giving the AI the option to continue writing the story for the author it allows the author to explore the story for themselves. This is something I always wanted to do, but given the fact that I wrote it, I do already know everything about it. I never had the chance to see my stories from a spectator's point of view.

And one last note: Never ever do I claim any AI-generated texts to be 'mine'. I do claim the additional training, fine-tuning and prompt design to be my work, but everything that comes after that was generated by the AI. Of course, it is hard to determine, whom the generated art of an AI belongs to. My opinion is, that it belongs to the AI itself the same way both of the example stories used in this thesis 'belong to me', but this is a field of discussion that is beyond the scope of this master thesis and probably even beyond my own knowledge.

I know, it may sound wired pointing all of that out, but events happened, that made me want to clarify that.

1 Introduction

1.1 Motivation

Many computer games are supposed to bring the player into a world beyond the real one and challenge them with tasks they usually don't have to face. Some games do an impressive job at this, some don't. Most video games try to achieve their goal by telling a riveting story, try the players abilities in reaction and precision, challenge them to solve puzzles or facing them with decisions which will have an influence on what is happening in the world or to the characters involved. The way this is done is often predetermined. For puzzle games and sports games, this is definitely no problem. Puzzle games for example are defined by the fact, that the player has a small set of rules and often limited set of abilities to solve them, which ultimately means there are only a few ways to do so. In sports games, the goal is for the player to find their skill level and rise to and beyond it. The fact that the end of such games is ultimately predefined (you either achieve the goal or you don't) does not really matter, since it is not the goal the player is interested in, but the way to get to such a goal.

In many games where the player takes over the role of a character of a story (role playing games or ego shooter for example) most of the time, the player is more interested in seeing the story unfold while the ways to get to such an unraveling is often pretty much the same. One may have some ways of deciding on a party they want to favor, or whether they would like to break in guns blazing or sneak around and open a window quietly, but in any case one only has a limited set of ways and sub goals to choose from.

This, of course, is because the developer of a game can only implement so many options. They just cannot account for every little, tiny detail the player wants to do different. Often, restrictions even have to be implemented to prevent the player from breaking the game or maybe even killing the main character, but that is a different topic. In any case, due to such limitations, if one wants to have complete control over what is happening in the game world and how to find a solution to a story line, they would have to leave the computer behind and opt to play games, that do not have such restrictions, for example pen and paper games (PnP).

Such games in most cases work like this: One person will be the game master (GM). They come up with a story which they present to the players. The players then do what they desire to do, while the GM has to react to that. The scale of the story and the world can vary from one GM to the other. If they only want to have a few hours of playtime, they may come up with a small dungeon and add a couple of traps and an enemy encounter. If they want to go big thou and develop a whole world that may provide material for years and years of story content, they are absolutely free to do so. While the GM is writing the story, the players develop their character. Again, the depth of the character depends solely on how much the players want. There is no limit. Once the story and the characters are prepared, the GM tells the players their current situation. For a small quick game the GM may just say: 'You are here, steal this treasure', but for a larger campaign, the characters may not even have a quest at the beginning and have to find their own problems.

The narration-based complexity and customizability of a pen and paper games stands in contrast to the challenge-based semi linearity of a computer role playing game. This, of course, is due to technical limitations of a software. A computer can not 'make stuff up' and therefore the developer of the software has to preprogram everything it is supposed to contain. For this reason, it is near impossible to bring such complex aspects of a pen and paper game to a computer game. However, with the recent development of artificial intelligence (AI) and their ability to seemingly come up with ideas, there is now a way to at least have a computer *seem* creative. Therefore, it may be possible to use such an AI to account for missing flexibility in a preprogrammed computer game and open new ways of playing.

1.2 Goals

The main goal of this master thesis is to find a way on how to use an AI in the context of computer games. For this purpose, a system called 'Virtual Game Master' (VGM) will be developed. Said system is supposed to be capable of recreating a gaming experience known from PnP as a single player computer game. The reason for picking a PnP is simply because these games already offer a great potential for creative content by design. Therefore, many opportunities for an AI to generate story relevant content are available right from the start.

VGM will be capable of containing rules and following a structured story line like a computer game would do, while at the same time being able to individually address the player's intents and react accordingly.

With this premise come a couple of main objectives the system has to fulfill. First, it needs to be able to react to whatever the player wants to do. Inherently, that means, the system has to provide a way to let the player do what they want and then interpret it in a way that can be further used in the system. Furthermore, an option for implementing a story or at least a setting has to be available. The player has to be guided through the story, while being free to alter it as they choose when exploring their own path. Therefore, the system has to provide ways of defining a story of any kind in a way, that it can be extended while playing. The type of story should not matter. Whether it is supposed to be a long-term story set in a fantasy world where the player is supposed to fight a war for the greater good or a quick and easy one shot in which the player just has to break into a vault and steal all the money, the system shall be able to capture as many situations and scenarios as possible.

To evaluate the viability of the system, a prototype shall be developed. The prototype will be split into two parts. The first part will be the rule set, which is necessary not only for keeping track of things like the player inventory or the overall state of the player. It is also supposed to keep things challenging for the player, for example by enforcing realism and a battle system. Furthermore, the rule framework is responsible for keeping track of what the AI framework, which is the second part of this system, does.

The AI framework is responsible for the creative part of this system. First, it is supposed to take over, when the rule framework is unable to respond to the player. This might happen if the player wants to do things that have not been implemented in the story. This means, it is supposed to write texts, that answer players questions and, if necessary, extend the existing content to fit the narrative. It is also supposed to generate texts that describe rules, for example for the battle system. Battles in a PnP game can be quite dull, since they often follow a pretty strict set of rules. What makes them interesting is how the whole fight is presented and flavored. In PnP games, this would be the responsibility of the GM. For VGM, this part will be taken care of by the AI framework.

Part I
Design

2 Analysis

Before implementing a computer in a system that is usually run entirely by humans, said system has to be analyzed. This can help to narrow down what parts exactly can be driven by which part of the system and how these parts have to communicate with one another. Furthermore, it will help to split the entire software system into several smaller parts with dedicated tasks.
There are various types of role playing games (RPGs), for example Computer RPGs, where the player controls a virtual character in a computer game, Live Action RPGs, where players actually take the role of their character and play out their actions in real life and Pen and Paper RPGs, where characters are represented by statistics written down on a piece of paper and everything that happens to or is done by the characters is communicated verbally. Given that most RPGs except for computer RPGs are driven by human parties, it is practically impossible to generalize and therefore scientifically analyze them. Although there are often rules to the game, they differ vastly from one system to another. Furthermore, the rules are not the only thing that define how an RPG works. In most cases, the players are granted a lot of freedom in their style of playing. However, a few theories about RPG have been written. These theories can be used to narrow down problems that may arise during the game and improve on them in later sessions. Two models arose and are generally accepted and used in the scene, namely the 'Process Model of Role-Playing' and 'The Big Model'. These models will be used to analyze Pen and Paper Game Systems for this master thesis.

2.1 Analysis of the System

The Process Model of Role-Playing describes role playing as 'any act in which an imaginary reality is concurrently created, added to and observed in such a manner that these component acts feed each other.' [Koistinen et al., 2005, p. 2]. This definition is strictly designed for the Process Model, nevertheless it already shows one of the main differences between a computer RPG and any other type of RPG. What makes PnP so interesting is the fact, that the imagined reality is not premade but created, changed and formed while the game is running. A computer RPG would not fit this definition, since for the most part, the imaginary reality is not created and only little added to. It has been pre-written and the player can now only explore it.

Before taking a closer look at the entire system, to avoid confusion, two terms have to be defined first. The space, in which the player explores the imagined reality and their expectations and hopes are placed in, is called the 'Imagined Space'. The overlapping Imagined Space of all participants, be it human or in the scope of this master thesis an AI, is called the Shared Imagined Space (SIS). Although the entire role playing system reaches beyond this SIS, for reasons explained later on, the structure of this space is the most important in the scope of this thesis. Since the SIS is imaginary, an entity is needed to enforce a common ground in which the role playing can take place. This entity is called the Authority over the SIS. In most cases, at least for PnP, this is the GM. As the name suggests, in the scope of this thesis, the authority of the SIS is given to VGM. This decision has several reasons, all of which will be explored during this thesis.

The Process Model and the Big Model both define three important parts of a role playing system. Like every system, an RPG has an agenda – a goal or desired style of playing in order for the participants to enjoy the game. Furthermore, a few components that work as different agents in the system can be defined. And finally, system resources the agents can use to reach the agenda, called Elements, can be found.

2.1.1 Agenda

The main goal or even the reason for playing an RPG, like with most other games, is to enjoy the process of playing. These processes therefore have to be designed in a way that they are enjoyable to the player. In terms of an RPG system, this design goal is called an agenda. Ron Edwards bases his 'Big Model' on the Threefold Model [Edwards, 2001]. This model identifies three main agenda for an RPG.

Gamism focuses mainly on the outcome of the game. Players participate to try to win. Therefore, often win and a loss conditions are defined, be it long-term or short-term. The strategies of the participants are designed to reach the set goal.

Narrativism on the other hand, focuses on what is happening during the game. Decisions of the participants are often not driven by what *they* think would yield the best result, but by what the character they are playing would do or what is the most likely event to occur within a certain setting. The goal is to tell a story while it is happening.

Simulationism focuses on how the game is being played. In contrast to Narrativism, the goal is not to tell an epic story, but to follow a certain premise as realistic[1] as possible and explore, how it plays out.

Edwards theorizes that every game, player, GM and world can be fit into at least one of these categories [Edwards, 2001, ch 2]. According to Edwards, these three models are exclusive for one instance of a game. It is, however, possible to employ aspects of a certain agenda that contribute to the goal of the main agenda. For example, a party following a Narrativism agenda would most likely fight in a way to win (Gamism) because who wants to see their character loose or even die. However, they would most likely still not have their character act differently just to be victorious. The narrative would still precede the game.

2.1.2 Components

According to the Process Model, role playing consists of four types of components, which interact with each other. From a system design perspective, these components can be seen as the types of agents of the system. These components are Results, Processes (social and role-playing), Circumstances and Methods [Koistinen et al., 2005, p. 3ff]. Figure 1 shows, how these Components interact with each other.

---

[1] Realistic in the scope of the imagined reality.

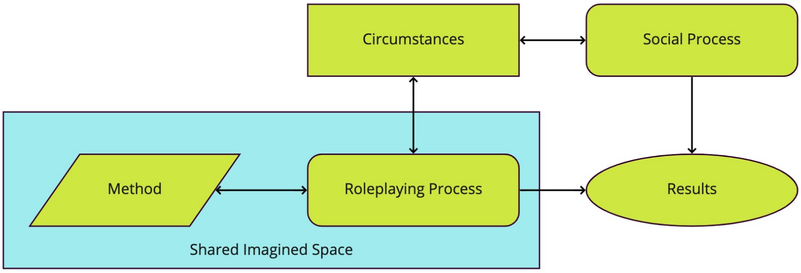


Figure 1: Components of the role playing system. (Source: own illustration)

Results are whatever comes out of the role playing game, be it desired or not. It is important, not to confuse results with the system agenda. While the agenda describes philosophies after which to design components of a system, the results describe the measurable outcome of the game. They are not necessarily the goal of the game. These results may be of course the entertainment the participants experience during the game. They also include, among others, learning, social benefits or even just aesthetic appreciation. Results can also be negative, like boredom or social dysfunction.

Processes describe what actually happens during a game. The Process Model requires Processes to have certain characteristics [Koistinen et al., 2005, p. 5]. Most importantly, it states, that whatever comes out of a process can always be mapped to a Result. Processes always produce something that measurably influences a Result. When using and designing certain processes, it is advisable to do so while keeping the overall agenda in mind. Having the players compete in a tournament for example is a process of player competition. During a play following a Gamism agenda, this would most likely work without any problems. In a play that follows a Narrativism agenda, however, such a competition would have to be designed in a way that it also motivates the characters to compete against each other.
Other than competition, processes can be suspense, immersion or exploration of entities or concept through the SIS.

Methods are the components that describe *how* something happens. In most cases, PnP enforce some set of rules that restrict the players in what they can and cannot do. Most PnP games, for example, use dice to determine the outcome of certain events that cannot be explained purely by role playing a character. Another essential Method of an RPG is the distribution of authority in the SIS. In most PnP games, the GM has complete control over the SIS. However, they may pass some of that authority to the players, if they choose to do so. Other methods of RPG systems include the way of playing (live action or verbally) or how to represent a character.

Circumstances describe everything, that may have an influence on Processes and Methods, but does not happen within the game. For example, the general mood of a player of course changes the role-playing Process, but it is not considered a part of this process. Other circumstances are for example the relationship between players or distractions within the environment where the play takes place.

2.1.3 Elements

The last part needed to describe a role playing system are its resources. In the Big Model, these are called Elements. They are needed to shape Methods and Processes and have a great influence on the options available to implement the agenda. Edwards narrows down five elements [Edwards, 2001, ch.1]

Character is a fictional entity, be it the player's character, a non-player character (NPC) or the big bad evil guy.

System is the way by which events that happen in game are being determined. A very common System Element is the rule set. This element has the biggest influence on how methods are designed.

Setting describes the current environment the character is in. This does not only include the location and time, but also things like their background story and events that happened in the world before the game takes place.

Situation is what actually happens within the game. The problem a character is facing or a quest that has to be done.

Color are the details that describe certain nuances of the setting to provide an atmosphere.

2.2 Analysis of Methods

Having outlined the components of a PnP system, the structure of them shall be analyzed in this chapter. For reasons explained in part II Results and Circumstances only play a small or secondary part in designing VGM. Therefore, a few decisions have been made to shape these components. These will also be explained in part II.
Processes are greatly shaped by the available Elements. Designing these elements, however, shall be left open to an author who can write a story that is being played with VGM. How these elements are being implemented and influence the Processes that can be used is greatly defined by the available Methods. Therefore, designing these Methods is the first important part when designing VGM. To get an idea of how Methods can be designed to create VGM, a closer look at the more frequently used[2] of them is taken. By analyzing these methods, decisions can be made to develop and implement possibilities within VGM.

2.2.1 Method of presentation

The first thing one will notice, albeit subconscious, is, that there is not one single style of refereeing a game. Every GM has its own style of presenting the settings, situations and characters. But after a while one may see, that there is a pretty quantifiable spectrum.
One method of presentation is to leave everything to the players. The GM most likely will present a problem that exists, but the players will have to find every clue and every piece of evidence themselves, forging their own paths (From here on out this style is referred to as 'open presentation'). This can feel especially rewarding for the players because they not only may or may not complete the quest, but they also came up with their own solution. No matter the end, they are responsible for it. The players will be way more involved not only in the world, but most likely also with each other. This is because Elements are not just presented to the players, they are also allowed to implement their own, especially Situations, Characters and Color. Therefore, the role playing Process is greatly shaped by the players themselves.

---

[2] At least based on observations made during games I have been a part of either as a GM or a player.

The problem with such an approach is, if absolutely no solutions are provided, if the task at hand is particularly complex like taking over a government or undermining an underground organization, players can become a little frustrated. They may spend hours on hours searching for a viable path, realizing nothing really would work out. Another problem, albeit for the GM and not the players, that may occur is, that certain story lines may not be discovered or even be destroyed. In one of my campaigns, for example, I planned for the players to board a ship and become part of the pirate crew seeking for mystery and treasure. Instead, one of the characters, while not necessarily having a problem with piracy itself, disliked the fact, that they also used to ship slaves. So instead of boarding the ship and opening a campaign of sea fairing and bounty hunting, they burned the ship to the ground, ruining 70% of what I had planned for the next sessions. However, since players form their own paths, when following a Narrativism agenda, this method of presentation may be the one to go for.

The other way of presenting a story is to completely guide the players through it. The problem is presented as well as one or the other way of solving it (from here on out, this way of presentation is referred to as 'railroading'). In most cases, the options will be shown for example by a dialog with an NPC or maybe the players find some sort of guidebook which already says how it is done, and they now only have to execute. In any case, a GM who is only railroading will always find a reason, why another solution will not work or talk their way around it until the story reaches one of the predefined paths again. In many ways, this is basically exactly how computer games work. One would put more effort into presenting the story and solution, but less effort into developing what is beyond it. Even thou railroading may sound like the less interesting method, it comes with some benefits. Most importantly: The players are getting things done. They spend less time preparing their character and rummaging for solutions and get to the nitty-gritty interesting puzzle solving and orc slaying part quickly.

Another upside of railroading is, that in contrast to an open presentation, the GM does not have to account for every problem that may or may not occur. This way, preparation time can also be shortened by a significant amount of time. The downsides of railroading on the other hand are pretty obvious: The players don't get to choose. If the game follows a Gamism agenda, this should be no problem, since the participants just aim to win the game. Railroading them towards a certain quest may just be enough to achieve desired Results.

The chosen method of presentation mostly depends on what the players and the GM like and which agenda they follow. In most cases, however, both methods will get mixed. In an open presentation, for example, it may be helpful to railroad the players a bit, if they get stuck.

2.2.2 Seeding

The players are not the only creatures in the world. There are other civilizations with their own stories, intrigues and what not. The GM is free to tell as much about them as they want. The more they present, the more populated the world will feel. It might also happen that the players come across an interesting story line that they want to get involved in. This act of implementing new elements that previously were unknown to the players is called 'seeding'. A tiny bit of story has been shown to the player, and they may or may not take care of this and develop it into a full-fledged campaign.

Seeding does not only help with filling the world with interesting stories and NPCs, it can also be leveraged for several things. For example, as previously mentioned, to railroad the players back to the original quest or the opposite, distract them from what they are currently doing. Every so often, it happens that the main quest the players are currently following is getting really bloated and over their head. They stumble from one fight to another, loosing tracks of what is actually going on and may not see any reward, which may become frustrating. Seeding a missing person case from a total stranger for example may help the players take their mind off things occasionally. They basically get a somewhat instant gratification after they fulfill the short side quest, and may even learn something along the way about the main story.

Seeding however does not come without its costs. If the players, for example, find a hidden lair, the GM may start describing what they find. They may say, the characters find a few books lying around. That, of course, means, the players may get curious about them. Even though the GM may only have placed the books to make the habitant of the liar look smart and give a clue as to what their profession may be, the players can read the book as well. But in contrast to video games, where books often are just a visual thing, it would not be really consistent if the GM said the players cant read the book. If the players insist on doing that, the GM has to quickly come up with what the book may contain. This again gives the possibilities to seed smaller side stories, tell the players about myths in the world, or even develop a whole new story about the original author. In any case, the GM is responsible for thinking of something that fits their narrative.

2.2.3 Make the rules interesting

A big part of any game, whether it is a video game, a board game or a PnP game is to challenge the player. To reach a state of flow, the challenges the players are facing have to match their skill level [Nakamura and Csikszentmihalyi, 2002]. To define the skill level of the player and their abilities, rules have to be written. Most PnP systems do have rules for testing whether a character can do, what the player intents to do (skill check) or how battles are held. Keeping track of these rules, however, can get quite daunting and may also scare off some players. Not only that, but especially during combat, sticking to the rules can get quite repetitive since in most cases depending on the class of the character, their actions are somewhat limited while battles can get quite long. It is up to the participants to use elements that make these rules interesting.

For example, a not too uncommon way of rewarding players is by special items. These items often give the character's abilities they usually would not be able to use with their current restrictions, be it due to their class or level. For example, one might receive an Amulet with the spell 'Feather Fall[3]' on it. One may also find a tiny, finicky trinket that looks and even functions like an umbrella. When opening it, the player realizes, that they feel lighter. Technically, both items are identical, they both reduce the fall damage taken by a player, but arguably, the latter seems way more interesting.

2.2.4 Designing System Elements

rule sets in PnP games work by defining the shape of some of the system elements [Edwards, 2001, ch 5]. Some rule sets only define very few elements. GURPS, for example, only defines the system itself. Others like the Dungeons and Dragons rule set define a lot more, like system, color and setting or even situations. The more elements are predefined, the easier it is to jump into the game without a lot of preparation. However, it also gets more likely that some of the elements need adjustment to

---

[3] This spell is from the Dungeons and Dragons rule set. It reduces the falling speed.

fit the narrative. In such cases, one could either pick a different PnP system, or just change the rules a bit. This ranges from minor changes such as certain items being more expensive since they are less common in the specific setting all the way through creating an entirely new subclass because no pre-made subclass really fits the narrative of how a character actually got to the point where they are now.

Generally speaking, the less System Elements are pre-determined, the more 'universal' the PnP system becomes.

2.2.5 Visuals

Most PnP systems do not require or even recommend not using too many visual representations of the game world. Usually, the GM is responsible for describing the world, pointing out the most relevant details, be it for actual story relevance or just to set the mood. The players are free to form the landscape themselves according to what the GM says, filling in the blanks with what they seem to fit. This also has the side effect of the world becoming even more alive without anyone having to do anything. For example, if you were to fill a large-scale city in a futuristic world in a computer game you will probably add some flying cars, lots of neon lights and nightclubs but also smaller details like trash lying around or manholes in the street. If the player wanted to interact with any of these, you could just not let them. Even though the player can *see* something, they would not be able to interact with it, clearly showing, such items are just here to set the mood. Arguably, such small details may on the other hand make the very difference in a PnP game. The GM would describe the overall picture with the flying cars and what not, but leave the details to the players. If they are in a tricky situation, they may now try to come up with a solution the GM could not even have imagined. If the player were to ask whether they see a fire escape ladder on one of the balconies on the street, the GM could effortlessly add this detail, letting the party escape by means they didn't even consider.

That being said, most systems do indeed make use of maps when it comes to combat. As previously mentioned, most combat systems are packed with rules to keep it balanced, and therefore it can become necessary to know exact distances and orientations of enemies, covers, escapes and other points of interest (POI). For such occasions, it is highly recommended to use at least a simple map on a paper to avoid confusion. However, that does not mean, that combat cannot be held without a map. Often times, the party may encounter only one or two enemies in a very confined space. Due to short distances, range measurements may become irrelevant, covers can be easily dodged and area of effect[4] (AOE) attacks may be evaluated on a fair basis.

2.3 Analyzing Existing Systems

With the development of more and more powerful software, computer games are also offering more and more possibilities to the player. Developing a system that lets the player's imagination run free and let them do whatever they want is not a new concept. To understand, how some of these games use their Elements and Methods and how they work towards or maybe hinder certain Processes, a closer look shall be taken at them. The games analyzed in this section have been chosen mainly because they excel in at least one of their core points. This list is by no means comprehensive and only serves as a starting point to analyze positive and negative elements and methods for such a system.

2.3.1 aidungeon

'aidungeon' is a project by a small developer team that uses an AI to tell a story[5]. The player can then describe what they want to do and the AI reacts accordingly. The game follows a Narrativism agenda. It lets an author write a setting, situations, color and characters. The player is then free to explore these.

There are a few interesting things about this system. First, the player is not restricted in what they want to input. They don't have to use specific words or a certain syntax. They can just type whatever they want.

I have played a few rounds with aidungeon. One of the dialogs can be found in attachment 12.1. At first, I just played along the AI to see the potential of the happy path[6]. The very first thing one will notice is that in most cases, the AI responses always somewhat relate to the user

---

[4] 'Area of effect' is a short name for an effect that covers an area larger than only one person. Examples are smoke grenades or the infamous fireball from the 'Dungeons and Dragons system.

[5] https://play.aidungeon.io

[6] In software development, the 'happy path' is referring to a scenario, where no errors occur. If the user is following the happy path, it means, they do whatever the software asks them to do in the correct way.

statements. The utterances are sometimes incorporated in the response, and said responses are not completely random. Interactions with NPCs also seem quite interesting. The NPCs often stay in their character and follow their given path. Another positive thing is, that the overall main story line is seemingly always present, albeit sometimes randomly. Sadly, this already concludes the positive statements that can be said. Quite often, the AI seems to change the entire situation the player is in. This mostly includes sudden location changes and wholly new characters the player has not even seen before.

Furthermore, it quite often happened, that the AI did nothing else than to respond to the player's input in an affirmative way but not stating any consequences that one might expect. In some cases, this may not be a bad thing, but especially when interacting with NPCs, the player will try to convince them to do something and then of course expects them to either follow through or bail. But when they just confirm the statement uttered by the player, nothing changes in the current situation. I as a player sometimes felt lost. The AI confirmed what I did, but did not say what happened then.

After realizing that it was pretty hard to get the AI to drive the story, I wanted to see how far I can go towards 'realism' and consistency. The character I created was a rouge. Therefore, I assumed that it was very unlikely for him to be able to use a lot of magic. The AI, however, had no problem with me using spells. I then said I wanted to transform myself into an ancient dragon, which at least in the Dungeons and Dragons Universe is considered a powerful creature. Transforming into one of them requires a very high-level wizard. Again, the AI played along and let me do that. But then it actually tried to challenge me by sending warriors against me. However, after I then stated that I am far too powerful to be killed, the AI just confirmed that.

A few things can be concluded from this play through. First, it is indeed important to give the player some sort of guidance as to what their mission is. It does not seem like aidungeon makes use of Methods that can help with railroading. Of course, it is not necessary to tell the player their next step or for the player to even follow them. However, at some point, they do need hints as to what they could work towards to.

Secondly, keeping an AI on track is never an easy task. This problem was very obvious in this game. The reasons for that will be explored in section 7.2.2. Therefore, there has to be a system that collects and keeps track of the necessary information to have the AI perform better.

The third thing is, that some system Elements need to be defined.

Aidungeon does not restrict the author of a scenario or world to a color, character, setting or situation, which leaves them total freedom in what agenda they follow. But it also does not implement a system. Although it can be quite fun to turn into a dragon and reign over an entire country, if there are no challenges in doing that, the fun will be over just as quickly.

2.3.2 Dwarf Fortress

In 2002, two developers started working on the game 'Dwarf Fortress'[7]. Over the past 20 years, it became probably one of the most complex games ever made. The concept, however, is simple. A world is procedurally generated and the player can choose whether to join that world either as a small group of traveling dwarfs sent out to build a new fortress, or as an adventurer willing to discover all the different cultures and mysteries the world has to offer. The world is persistent, meaning, the player can change their role or start a new colony whenever they like. The game does not offer a story or quests per se. It has a purely Simulationst agenda. It simulates what happens within the world with great detail and have the story write itself. What makes this game interesting for this master thesis, however, is the sheer complexity the game has to offer.

In short: everything that is being done in the world will have a consequence. For example, if a dwarf orders a beer in a tavern, the bartender may take a moment to actually fill the mug and serve it (bartender behavior). If the ordering dwarf, however, had to attend a job, they would then just chug the beer, covering the floor in fluid (dwarves behavior). Of course, a puddle would form (fluid behavior). It could happen that a cat would walk over the puddle and now being dirty would start to clean itself (creature behavior), therefore consuming alcohol and actually getting drunk (toxins behavior). The entire chain of the events, however, is not preprogrammed. It is just the result of how different parties act in certain situations. By having a modular system that allows actors to be controlled by events happening to them, it is very likely that things happen the way one would *think* they happened instead of how the developer originally implemented them.

Dwarf Fortress offers a great look in how realism can be implemented in

---

[7] https://www.bay12games.com/dwarves/

a game. It is important for the system to describe *how* elements work and not what they can do to other elements.

2.3.3 Scribblenauts

In the game Scribblenauts[8], the main character can summon objects just by typing their name in his notebook. Furthermore, he can use adjectives to influence the properties of the summoned item. It follows a complete Gamism agenda. Although a background story is implemented, it hardly ever matters during the game. The player has to collect points by helping people. The game hereby uses item properties to which other elements can react [Schreier, 2017]. The player may for example find people standing on top of a building, claiming they have lost their way and want to get down from there. The main character could create a plane or a helicopter and let them fly down, but he could also summon a dragon and give it to them as a pet. Both actions fulfill the quest, since both items can *fly* and can be *mounted* by passengers. However, there may be a few people in the city who are afraid of *dangerous* creatures. Furthermore, the police man in the city will attack everything that is *aggressive*. Therefore, a much more peaceful version that also includes animals would be a giant bird.
By making use of properties, all items roughly behave and act as the player may expect, although the core principle of the entire game is pretty straightforward: summon an object that has a certain property. Scribblenauts shows a good way of implementing behavior of elements. The interaction between them is not predefined, they act themselves. However, the mode of communication, in this case the different attributes of the elements, allows for a way more complex interaction.

3 System Design

Using the observations from section 2.2 VGM can now be designed. In this chapter, all design decisions made for VGM are being explained. Based on these decisions, a system structure is proposed. A brief overview and explanation of communication between the system agents is given at the end of this chapter. Overlying premise of almost all decisions, however, is the fact, that VGM is not only supposed to be a game to be played with a predefined setting but also allow authors

---

[8] https://store.steampowered.com/app/218680/Scribblenauts_Unlimited/

to implement their own settings and have players play them. In other words, first, the author creates the SIS and allows the player to explore it. Then the AI learns how this SIS is created and adds to it.

3.1 Agenda

VGM is supposed to follow a Narrativism agenda. This decision is mainly based on one observation[9]. As seen in section 2.3, implementing Gamism and Simulationism agenda is a task, that can be achieved only by using 'normal' software. An AI is not necessarily needed. The reason for this is, that these agenda follow a very calculable path. Each input always has a somewhat predetermined output, be it either for example due to a strictly designed score system for Gamism or specific element behavior for Simulationism. These goals somewhat prevent the use of an AI, since they are designed to work creatively and therefore can become really unpredictable. However, to fully explore the key premise, implementing an AI in a computer game, a Narrativism agenda would leave more options open.

Another reason for picking this agenda lies within the fact, that VGM also shall allow authors to implement their own stories. Depending on the author, they may follow a different agenda than Narrativism. However, finding a way to implement all three agenda may overcomplicate things when it comes to designing ways of communicating with the AI. Nevertheless, in his Big Model, Edwards states that '[...] all three models are social applications of the foundational act of role-playing, which is Exploration.' [Edwards, 2001, ch 2] Even if it's just 'exploring the options', focusing on that aspect of the game may allow for other agenda to be implemented by an author. While Gamism and Simulationism agenda mainly focus on exploring a few Components and Elements of the entire game, personal observations show, that a Narrativism agenda often provides the most space for exploration while at the same time allowing for an interesting story to be implemented.

3.2 Elements

When designing an RPG system, what usually is done is to define how certain Elements or Methods are supposed to work within the system. Depending on which Elements are predetermined and which are left open for customization, Methods, Processes and even the Agenda can

---

[9] Aside from personal preference

be adjusted at will. Generally speaking, the less is defined by the rules, the more 'universal' the role playing system becomes.

Given that in VGM an author is supposed to implement their own story, it makes sense to leave Color, Settings and Situation up to them. But especially when it comes to the two latter, these are also key Elements for the AI to be written by. By having an author write these three Elements, an AI can learn from them on how to design new Settings and Situations.

Characters, on the other hand, are a more complex topic. They have their own Settings, can create Situations and also interact with each other. During the development of VGM, plenty of tests have been conducted to find good ways to implement characters in VGM. Sadly, no solution has been found that works with the overall design philosophy. More on this problem is explained in section 9. For now, however, the focus has been shifted towards the first three elements. The player character is predefined and NPCs are left aside to keep variables to the part that actually matters in the scope of this master thesis, namely the development and training of the AI.

All this leaves one Element open, this being the System itself. This includes rules for a battle system, basic physics and navigation, character stats, skill checks and item behavior. Since these Elements are crucial to design a challenge for the player, due to the unpredictability of an AI, it would not be feasible to have an AI decide on them. Therefore, the rules for all these are being pre-defined by the rule framework of VGM.

3.3 Interface

Traditionally, PnP games are played by verbal communication. Pen and Paper are only being used to write down character stats. Of course, it would be possible to use voice recognition and voice synthesis to produce the communication between player and VGM. However, this may pose a few problems.

First, voice recognition is still susceptible to misunderstandings. Common words will be recognized fairly easily, but when it comes to made up words or names of characters or locations, the recognition will run into problems. If the voice recognition failed to understand something, the player would only realize this after receiving VGM's answer and realizing it would not fit.

When it comes to voice generation, these as well are not yet on a level where they could be used in such a context. There are several voice generation models out there. A basic text to speech synthesizer (TTS) is provided by operating system developers like Microsoft [Urban, 2022] or Apple [Apple Inc., n.d.]. The API of these allow to use TTS in almost all software. They, however, are mainly meant to be used as an accessibility feature to help visually impaired people read. They are designed to 'read' texts quickly without waiting too long for the audio. These software programs do not analyze the meaning and words of a text, they just put the general pronunciation of syllables one after another. Therefore, they sound somewhat boring and would not fit the framework that is supposed to read interesting stories.

Other models are trained a bit finer. They do analyze the general meaning of a text by looking at word positions and punctuations, and manage to pronounce the speech in such a way that it sounds somewhat natural. However, depending on the model, the time for generating the voices is very long. Considering the fact, that the AI responsible for generating the text already takes a few seconds, adding even more waiting time to generate the voice would break the flow of the game.

Instead of using voice generation and recognition, the entire system shall be developed text based. This still leaves the option to add voice generation and recognition later on. Furthermore, only using text to communicate the story allows for more freedom for the author to design the Color of the system and the Processes used. Especially the latter will benefit from the fact that neither the author (and therefore the AI) nor the player are restricted in their respective texts by predefined input and output methods.

Since the interface still has to represent some sort of communication between player and VGM, it is designed to look like a basic text messenger as known from smartphones. Using such an interface brings the advantage of not having to explain how the game is being played if the player is already familiar with them. Furthermore, these interfaces follow a lot of the design principles developed by Jacob Nielsen [Nielsen, 1994]. By only allowing one single method of input, there is technically no possible way for the player to make mistakes (Error Prevention). If, however, VGM is unable to interpret what the user entered, these problems can be stated directly by VGM in plain and explanatory text (Recover from Errors). The interfaces do not contain unnecessary information (Minimalist Design). Another advantage of using such a design is, that the text messenger counterpart almost always comes with an indication that shows the user if the person they are texting is currently writing something. Incorporating such indications in the interface design of VGM allows for covering the time the AI needs to generate an answer. This is useful even if the AI does not need to do that. Sometimes, the answer can be found in the pre-written content. In such cases, no time is needed to generate an answer. However, sometimes VGM splits the answer into multiple smaller texts. Play tests have shown, that if they pop up all at the same time, players just started reading the very last message, wondering why it is out of context before realizing that other messages have been sent previously. Artificially delaying messages and showing them one after the other while at the same time showing an indicator that looks like someone is typing changes the feel of how the communication with VGM is done. It helps the player to keep track of the messages that are visible and also indicates, when VGM is waiting for the user input[10]. (Visibility of System Status).

3.4 Frontend

Designing the frontend of VGM as a web-based application mainly has two reasons. First, such applications can be run on almost all devices. The clear advantage of that is, that the Circumstances of the system can be chosen by the player themselves. They are free to play the game when they want to. The restrictions on accessibility to devices that can play the game are being cut down, since almost every modern device can run web-based applications.

Secondly, since web-based applications make use of the Cascading Style Sheets (CSS) the author can be given the option to change the look of the app by just writing a simple text file. This allows for further control of the Color of the system by the author. If they want to write a medieval fantasy setting, they may opt to use brighter colors and more playful fonts, while when writing a modern horror setting they are free to use the opposite.

4 Overview of VGM

Based on these design decisions made in section 3 the following system structure has been developed and will be explained in more detail in

---

[10] Although it is programmed to not have to ’wait’. If the user inputs something, it will always be processed chronologically in accordance with what is being done by the system to avoid situations that feel like VGM was not paying attention to the player and therefore the player having to repeat themselves.

part II. There are three main components of VGM with that being the Content, the rule framework and the AI Framework. All required System Elements (Color, Situation and Setting) are being defined in the Content by an author. Based on the Content, the rule framework interprets the player's input and evaluates, what is to be done. It then collects all necessary information needed for the AI framework to do its job. The training of the AI then completely depends on what has been defined by the Content and how it is structured. After the AI generated its response, it's being output. Furthermore, an opportunity for static Content Elements like Events to be displayed is provided at this point.

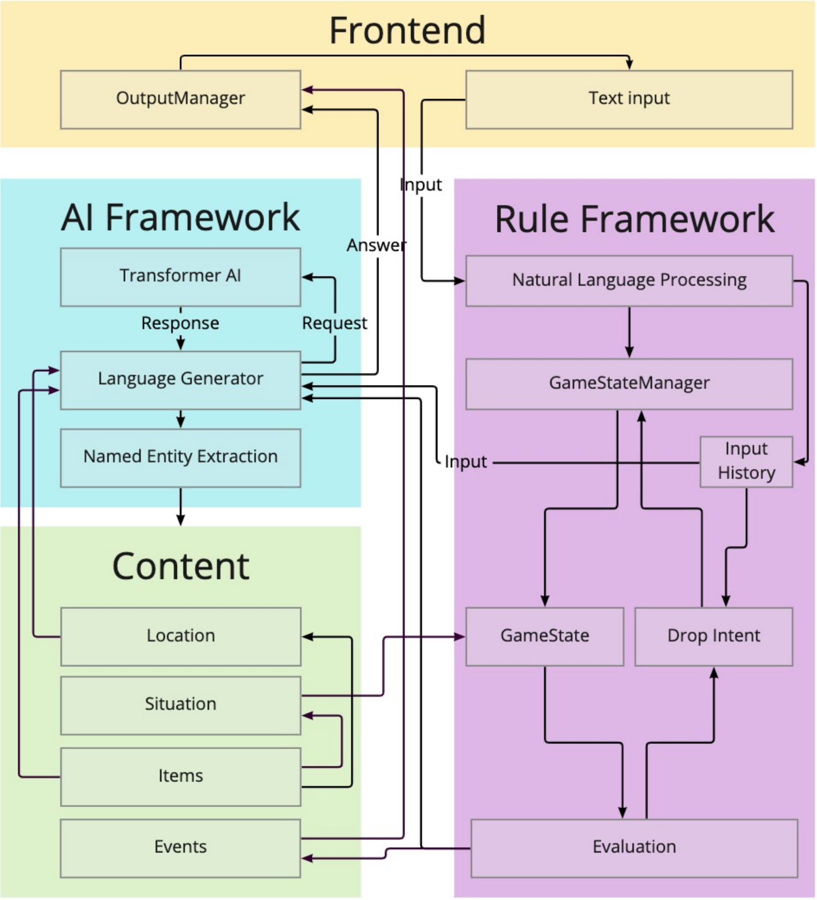


Figure 2: Flowchart of the VGM System

Part II

Development

5 Content

To get an idea of what the rule framework has to be able to do and which Elements have to be represented in the Content, the story 'Risas Secret' [11] has been written. It can be read in attachment 12.2. This story has been developed in such a way I, personally, would use as a basis for a PnP game. First, it contains multiple endings, which can arise depending on the player's action. Some of these paths can also open or close themselves over time, for example if the player were just to do nothing, and wait for two days, the worst ending would happen. All these paths contain several aspects one may find in a usual PnP game such as conversations with NPCs, searching for items, finding unexpected item and exploiting them, solving puzzles and ultimately a fight against the main boss. Furthermore, some events have been implemented to grab the attention of the player to show new paths or help them out a bit. The content allows the author to write Situations, Settings and Color that describe the world and what may happen to the player.
The content includes descriptions of locations, events, situations and items. These are called Content Elements[12].

5.1 Locations

Locations describe an environment the player can enter. These do not necessarily need to be confined spaces. A location is mainly supposed to give the player a description of what they see if they were to think of that environment. Therefore, a location can range from an entire continent to a broom closet under the stairs. Even POIs like the well in the middle of the town can be considered locations if the author of the world intents for the player to interact with it. The main aspect of locations is, that they are connected to each other. Exits describe other locations, to where the player can go. Using such a system, allows restricting access to some locations. This is significant not only

---

[11] The author of 'Risas Secret' is myself.

[12] Not to be confused with System Elements.

for locations like Risas hut from the example story, which of course is locked, but also allows for accessing locations in several ways. In the example story, the player may see a purple glow coming from the well. Some players may decide to climb down the well. This, of course, is no easy task, and they may want to use a rope to make the descent a bit simpler. These two possible options would be implemented as a so-called exit situation. The exit describes several ways to be used which automatically triggers a situation as described later on.
Another benefit of structuring locations in such a universal way is that they later on can be easier replicated by the AI. Not having to differentiate between certain types of locations significantly reduces training and prompt design.

5.2 Items

Items are everything, the player may find and use during the story. In many classic computer games, items that are used for certain things are implemented in such a way that, if the player carries a certain item, they can do what the developer intended to do with said item. Often, one item serves one purpose. In a PnP game, however, this is not a desired behavior since it restricts the options of the player. For example, if the player wants to go down the well mentioned previously, they may seek for a useful item. Said item does not need to be *a rope* or *a ladder*. It technically only has to be an item that is long enough to reach the bottom of the well. If the player was down in the well, they definitely would need some light to find a way. But they do not need *a torch*. They would need something that emits light and is bright enough. Using these considerations, items are implemented similarly as done in 'Scribblenauts' explained in section 2.3.3. They have certain properties. When checking conditions for situations, these properties can be checked for requirements instead of naming certain items needed. Furthermore, this approach allows having the AI generate and even modify items in a way the rule framework can understand. This way, if the player were to pick up items that originally were not meant to be used by the author, an AI could generate the most important properties. The rule framework then saves the new item and can then use it in the story.

5.3 Situations

Whenever the player is facing a decision that has to be made to advance in the narrative, a Situation[13] is created. A situation that would have to be written in the content would be the aforementioned entrance of the well. The author would describe the requirement to continue to enter the well. In this example, it would either be a basic skill check to see whether the main character can climb down without any problems or the use of a certain item of a length, which would make the skill check easier. The situation then describes the result. In this case, if the player decides to take the action and not back out, they will definitely have changed the location to the bottom of the well. However, depending on the result of the skill check and the use of the item, they may or may not do so unharmed. They may even fall unconscious and only wake up the next day.

Situations like that can be leveraged to describe puzzles like entering a certain key code for a safe or more complex location changes, as just mentioned. Of course, it would be tedious to manually write a situation for every single door that is locked and just needs a key or a lock to be picked. Therefore, the rule framework has been written in such a way that if a requirement has been specified for an action, but the result is somewhat self-explanatory (like opening a door), the situations will be created automatically.

5.4 Events

If something happens without the player having an influence on it, this is considered an Event. They may be triggered by something the player did, but they just happen no matter how the player reacts. The purple light seen from within the well, for example, uses an event. They can be used to catch the player's attention and to describe things that happen in the world. They also can change some things and trigger new events. Furthermore, based on what the player does, they may trigger single messages called notifications. These can be used to describe events closer based on where the player is located and how they interact with the aforementioned event.

Events can be used to create an atmosphere or seed plot points and

---

[13] To avoid confusion between Situations as a System Element and Situations as a Content Element, if not clear in context, these Situations are referred to as Content Situations.

distractions. After all, the main character may not be the only creature in the story. Using events, doings of other creatures can be described based on what the player is doing. Furthermore, Events can be used to guide the player's attention towards certain things. In 'Risas Secret' for example, the purple light the player sees is an event, since it happens regardless of what the player does. Notifications are then used to inform the player where the light is coming from as soon as they get closer to the well.

6 Rule Framework

6.1 Objectives

The main objective of the rule framework is to keep track of everything that is happening in the SIS and to enforce the System. Changes that happen in the world will be written to the content, and the state and location of the player is recorded. The rule framework has to describe whatever the player can see that is already pre-written in the story. It has to recognize what the player intents to do, decide how this intent will be realized in the player's current situation, and pass that information to the Natural Language Generator.

6.2 Natural Language Processing

Whenever the player states something, the rule framework has to recognize, what exactly the player intents to do. This process is done by the Natural Language Processor (NLP). The NLP tries to understand the player's intention based on what their input is by classifying the utterance. It is also capable of detecting certain useful elements in the input. Although this objective is already being realized by a neural network, it is not considered part of the AI framework because generally speaking, a neural network (NN) alone is not an AI, but an AI may consist of one or several neural networks. The NLP is a classification network called nlpjs by the AXA-Group [AXA Group Operations, 2022].
Classification networks work in such a way that they take a set of inputs and calculate one of several most likely pre-defined outputs [Kinsley and Kukiela, 2020]. The most well-known example of this is image classification. A picture is given to the network. Depending on the construction of the network, the image is being looked at in various ways. Based on the results of the observations, the network calculates which of the

given output classes this picture will most likely fit in. NNs in general are excellent at finding patterns and regularities. Therefore, classification networks are usually easy to train and are very reliable.

Because the training material is provided in a way that the inputs always point to one specific output, training such a network is also a pretty straightforward task. The training material is already 'correct' and therefore the training process does not need human intervention. However, for that exact reason, training material has to be generated manually. It has to be assured, that the material is indeed correct. For VGM, this can be done in a simple way. Nlpjs only needs a few utterances the player may state during the game, and then the network is told how these have to be classified. Training material can also simply be gathered during play testing. On training, the NN then randomly picks these utterances, classifies them and calculates if it was correct. If it wasn't, it adjusts some of the internal parameters and tries again. The training is considered complete, if the mistakes the NN makes are below a certain threshold. Even with quite a lot of material to go through in the current state of the VGM prototype, this process usually takes less than a second. Therefore, it can be done on startup without considerably increasing loading times. The benefits for following this approach will be explained in section 6.3.

Classifying human language is not too difficult a task, if the emotion does not have to be considered. For most tasks, one would use the same or at least very similar words. If the player wants to pick something up, commonly they use the words 'pick', 'grab' or 'take'. But when it comes to things like using items, it cannot be broken down to specific words, since different objects require different verbs. A candle is being lit, a ladder is being climbed, a book is opened. A human GM would know, that these objects are things that are considered items. To make the recognition for the NN easier, another element can be used, and this is where the previously mentioned creation of content comes into play. Nlpjs can detect certain words and classify them separately. To achieve this, the network is being told that 'candle', 'ladder' and 'book' are considered 'items'. The training material is then changed to not use the words 'candle', 'ladder' or 'book' but only 'item' instead. The NN then reads the sentence and detects a word that is considered an item. It then already can narrow down the classification to intents that require an item. This process is being called 'named entity recognition' (NER) [Mohit, 2014]. This way names of locations, items, actions, numbers, directions, creatures and other entities can be implemented quickly. By

depending on a recognized entity instead of the name the player gave said entity, the reliability of the classification can be increased.
The classification and extracted entities can then be passed to the Game State Manager. The information includes several likely classifications including the network's confidence for each of them, the detected entities including their name and the original player's utterance.

6.3 Game State Manager

When the player utterance has been analyzed, classified and entities have been extracted, the game state manager passes this data to the current game state. These game states mainly contain the information, what intent has to trigger which action. Usually, the player is in the main game state. This state can interpret all basic intents like changing locations, closer observing POI or picking up items. More specific intents that occur when triggering Content Situations are covered in a Situation specific game state. Using this method brings a few advantages.
First, when the author pre-writes certain situations like for example when the player wants to enter the well in the example story 'Risas Secret' these situations can be introduced in a way that the player understands they now have to decide. The author can decide themselves whether they want to clearly state which options the player may take or hide some paths to make it more interesting. They then can define which path leads to which result. In the current example, one of these paths is entered, when the detected intent is a *location.change* like 'I want to jump down the well' or 'I just climb down' The other path is used if the intent *item.use* is being detected like 'I let the ladder slide down the well'. The main game state would just change location or state that VGM does not know why the player wanted to use an item. In the situation specific game state, however, both intents can be combined with additional checks like item requirements, skill checks or other things that happened or have been done. This way, every single situation can be adjusted at will. Most importantly, defining these situations can be done in a way that the author does not have to write extra code. They can purely focus on the description. This also means, that eventually an AI can be trained based on Situations written by an author, to generate them.
Another reason for choosing this approach is to help the NLP. Even though it is quite reliable, it is not infallible. It may happen, that the

NLP detects the wrong intend, especially if the player uses words, it has not been trained on. Many intents can also be quite similar. If the author intents to hide a secret door and wants the player to pull a certain lever, the situation would have a path that is being activated, if the player wanted to do that. However, the NLP would also have to learn this intent with new training material. Sentences like 'I want to pull the lever' are likely to be in said training material. The problem is, that such a sentence could also wrongly be detected as an intent to use a certain item, since this exact sentence uses an entity, that could be detected as an action ('pull'), and an entity that could be detected as an item ('lever'). This can lead to major confusions, if the NLP would always know that an intent such as 'use lever' exists. Separating situations and the main game states is useful to solve this problem. When the game state is being changed from the main to a situation, due to the rapid learning of the NLP, new intents can quickly be learned, and the ones not needed for this situation can be 'forgotten'. This way, the NLP would be more confident to classify such utterances in Situation relevant context. This will decrease false positives. If the player, however, states an intent that is not used for the situation, this intent can still be passed back to the main game state and be interpreted as it usually would. This process is called Intent Dropping.

7 AI Framework

Analyzing and quantifying human speech and generating results based on this information is a task that is easily achievable by a computer. Communicating results, which the rule framework calculated, in a natural sounding way, however, is not. Even if all the correct information is given, just generating a grammatically correct sentence can become quite complicated. Furthermore, it may quite often happen, that the rule framework does not *have* an answer, simply because the player goes off script for example if they were to break down a door instead of looking for the key. Or maybe the player suddenly is involved in a fight and this battle has to be described.

In such cases, a lot of the Methods described in section 2.2 have to be applied. Since most of these, like Presenting and Making the rules interesting, work with human language and some of them, like Seeding, are even meant to break the script, it is not possible to use common computer algorithms to do these tasks. This is where the AI framework comes into play.

7.1 framework Objectives

The objective of the AI framework is foremost to take care of generating texts that sound natural. Furthermore, the framework has to come up with texts, if the rule framework fails to produce an answer based on what is written in the content. Another objective that mixes the two former together is generating texts that explain and describe certain rules and uses of such in a way that they sound interesting. For example, a fight basically only consists of determining whether an attack or a defensive move is successful, calculating the damage if necessary and then adjusting the remaining health points of the damaged party of the fight. It then is up to the GM to tell the player how much health points the enemy has left. Some just straight out will say the number. However, some may also just describe the condition and behavior of the enemy, to give a rough estimation of how hurt the enemy is.
To develop means of using a text generation AI and determine how prompts can be formed using only the user input and content, a new example story 'Haunted House'[14] has been written. It can be read in the attachment 12.3. The first thing one will notice is, that this time, no NPCs will be used for the story. For reasons explained later on, implementing a character in a story is the most complex part of the entire system. Although methods of using an AI in order for VGM to play NPCs will be evaluated, they will not be part of the prototype. Other than that, the new example story takes place in a way smaller context, inviting the player to ask more detailed questions that will most likely not be answered by the rule framework. This story again has multiple paths to find the end, all of which can be used in several ways, making use of item properties and changes of locations. Furthermore, the horror like setting of a haunted house can be used to evaluate whether events have the desired effect of creating an atmosphere and keeping the player on edge and grabbing their attention.

7.2 Development

When it comes to AI that is supposed to be used for text related tasks, the choice would almost always be text generation AI. These work in such a way that they first read the given text and then try to continue it.

---

[14] The author of the story 'Haunted House' is myself.

Of course, there are several models out there, that are specially trained to do more specific tasks like summarize text, answer questions that have been asked for given texts and fill blanks in texts. However, such AI all work on the same principle. They first tokenize and transform the text. Afterwards, depending on their architecture, they produce an output based on the transformation [Rothman, 2021].

To design prompts for the AI that return reliable results, a closer look at how the AI work is taken in this section.

7.2.1 Tokenization

Tokenization is the process of splitting the input text into smaller parts or tokens. It would not be feasible to only split words based on spaces, since most languages change some part of the word based on the meaning of it within the sentence. Verbs in the English language are a good example for this. For a human, it is clear that the words ’call’, ’called’ and ’calling’ mean the same thing, they are just used in a different time frame. If the AI had to learn based on spaces, this would mean it would have to learn this exact word three times. This would be very ineffective and increase the learning time by quite a bit. Therefore, in general, a tokenizer would split these words into tokens like ’call’, ’ed’ and ’ing’[15].

The set of tokens can also be extended to include special tokens. These special tokens can be used to ’tell’ the AI certain things. For example, AIs that have been trained to fill blanks in texts most likely will be trained to search for a special token like ’<MASK>’ to replace. Similarly, AIs that only are supposed to generate short texts and have to learn the nature of these texts will be trained on several example texts, separated by tokens like ’<ENDOFTEXT>’ [huggingface, 2020].

7.2.2 Transformer Neural Networks

Transformer Neural Networks (TNN) are designed to process sequential data, like language. Other than Recurrent Neural Networks, which do process such data as well, TNN process the entire data simultaneously. This way, these networks can extract the meaning of each token depending on their position in the sentence. For simplicity’s sake going forward it is assumed, that one word is one token, although, as previously mentioned, this is not the case.

---

[15] Depending on the tokenizer, it is likely that it adds characters to indicate, that ’ed’ and ’ing’ are not the beginning of the word.

When a human hears the sentence ’A bird spreads its wings and flies away.’, given they know the English language, they understand that for example the word ’its’ refers to the ’bird’ and therefore the ’wings’ do as well. They know that ’flies’ is something, that ’bird’ is capable of doing, and that ’away’ describes a location or direction that is not ’here’. The letter ’A’ alone tells the human, that only a single entity of ’bird’ is being addressed in this sentence. An AI, however, has to learn all these relations from scratch.

An AI is trained by first trying to relate the tokens to one another. A completely untrained AI would of course not be able to do anything with the example sentence. But the more an AI is trained, the more it would realize, that the token ’bird’ and ’fly’ are used a lot together and therefore learn, that they are related. The same goes for the token ’wings’. The AI assigns ’meaning’ to tokens based on their relationship to other tokens (attention) and their position in the sentence (positional encoding).

In some cases, however, one word can have different meanings. The word ’key’ for example could be something to open a lock or a button on a keyboard. To differentiate between the two, the TNN pays attention to the tokens surrounding the token in question. It will eventually learn that one key relates to a lock and the other key to typing. This method is called self-attention [Giacaglia, 2019].

This is also where one of the more commonly known problems with neural networks comes into play. While they do relate tokens to one another, they can only look at so many tokens at the same time. Therefore, they tend to ’forget’ some bits of information, that have been mentioned too far back.

Nevertheless, using the methods of attention, self-attention and positional encoding, TNN manage to ’learn’ a certain language. They do so by understanding, how it is usually structured.

7.2.3 Prompt Design

For the NN to understand, what texts it is expected to generate, in simple cases an easy question might suffice. However, given that the player can enter whatever they want, just passing the player’s input to the AI may not be useful. Especially since the AI may need additional information from the content. Therefore, the input data has to be structured in a specific way. This is done by writing a lot of example texts that include an exemplary output the AI has to give. For example, if an AI has to write a stage play in the style of Shakespeare, the best approach to do so is to have the AI read just that. It would then learn how these stage plays are structured, what style of language is to be preferred, and what topics may arise the same way it learned how language works explained in section 7.2.2.

For VGM in particular, each field that the AI has to generate texts for has its own set of example prompts. These prompts are designed in a way, that they include all necessary and available information before an example response is shown. This way, the AI learns, that the prompt it has to generate has to relate to the previously provided information. This method ensures, that the AI does not suddenly invent Elements that have not been mentioned. How these examples look like will be shown in section 7.3.

7.2.4 Generation

Based on the model and how it has been trained or prompted, the response text is generated by the AI 'guessing' what tokens will follow the given text. Models like BERT have to be trained for a specific task and generate their answer accordingly, similar to for example a classification network. An input has a specific output. The reason behind this, is that BERT models are considering the entire text at the same time when calculating attention and self-attention. Therefore, they always know the entire context of every single token in the sentence, which makes their predictions extremely precise. However, this not only means, that a lot of training time is needed, it also results in somewhat predictable and uncreative answers. While this is helpful for tasks like Named Entity Extraction and Intent Classification, this model is not suitable for most of the tasks the AI framework of VGM has to perform [Devlin et al., 2019].

The GPT Model, however, does not necessarily need fine-tuning. These models calculate attention and self-attention the same way, humans would. They only relate tokens backwards, to what they have already read and then guess, what comes next. These models already are trained in a way that it tries to predict text [Brown et al., 2020]. If using a BERT Model, for every field the AI framework has to generate an answer, a new model would have to be trained. Using a GPT Model, on the other hand, allows to only use one model and design the prompts as explained in section 7.2.3.

7.3 Implementation

With an understanding of how the NN generates text, the Natural Language Generator can formulate prompts based on the task requested by the rule framework. By the time of writing this master thesis, the AI framework is used for four different tasks. These are Response Generation, Battle Description Generation, Content Creation and Named Entity Extraction.

After the Natural Language Generator received the request from the rule framework and recognizes a situation in which it needs to request a text from the NN, it gathers information based on the task at hand. It then sends the information to the server the NN runs on. The server then collects example prompts that have been designed beforehand, as explained in this chapter. After collecting and selecting the training material, the prompt is forwarded to the NN to have it generate the response.

In general, a good practice to follow is to design prompts in a very descriptive way. In the case of VGM, each line of the prompt is given meaning by directly stating its purpose. Since TNN have been trained to understand natural language, this way of prompt design is way more efficient than trying to assign abbreviations or numbers for the NN to make sense of. This way, the TNN can not only extract the meaning of each of the given lines in a prompt, but also understand what these lines are supposed to do. If it then sees an empty line, which it will, when it is asked to generate something, it can easier understand the purpose of the generated text.

7.3.1 Response Generation

Play tests of the pure rule framework without additional help of the AI Framework have shown, that the most frequently occurring situations, where the Rule Framework could not answer the statement the player was uttering arose, when they wanted to know or do something that was not mentioned in the content. One player, for example, asked whether they could see furniture in the corridor of the house in the example story. The content describes the corridor only as narrow and mentions a rug. Therefore, the player does not get the requested information by just showing the pre-written content again.

The play tests have shown, that such situations were easy to detect. Either the rule framework was not confident in classifying the player utterance, or it classified the utterance as a request for a description, although the current situation has already been described. In both cases, the player just wanted to get additional information about something that was just mentioned. These scenarios are categorized as 'response generation'.

Tests have shown, that a reliable and interesting answer could be generated by providing information about the location, a description, what the player can see, additional information that the player may not have received but is relevant for the story and of course the player's statement itself. One of the texts the AI received as an example looks like this.

```
location: corridor
observation: A narrow corridor without any windows. A well
    trodden rug dresses the floor.
prompt: Is there any furniture?
answer: No, other than the carpet, the corridor is empty.
```

The NN receives some of these examples. The last text it receives then is the one that actually needs a response. In this case, of course, the text stops after 'answer:' This way, the NN recognizes, that some text is missing from the prompt. According to what it has learned from the previously seen example prompts, it then starts writing the text it thinks could fill the gap the best.

This particular example shows the NN that it does not have to generate a simple yes or no answer if asked a question. It shows, that it can use elements that have been used in the line 'observation'.

7.3.2 Battles

In most PnP-Systems, battles are the one thing with the most rules. This is because they are supposed to be challenging. If the player could just do what they want, it would become boring pretty quickly. Therefore, in most cases during battles, the player just state what they intend to do according to their possibilities, e.g.: they attack with one of their available weapons and then follow the rules to determine whether they were successful. Afterwards, they may or may not flavor their attack. How exactly does the character attack, what do they do, how do they use their surroundings, do they taunt the enemy? But the basic attack stays the same.

In some special cases, however, it is the opposite. It may also happen that the player wants to do a specific thing and the GM has to determine which rules apply to determine whether they were successful and how the success or failure plays out. For example, if the player said they wanted to collapse a house wall to bury the enemy, the GM may ask them to make a strength check to see if the character is strong enough to do so. Afterwards, if they were successful, the enemy may suffer some bludgeoning damage and be restricted in movement for the next turn or until they break free. But even in this case, the player's utterance is 'just flavor' for the rule that is being applied.

Given this observation, the role of the AI framework slightly changes. It does not have to 'freely respond' to whatever the player said. This time, it has to 'make the rules interesting'. Since the player describes their attack themselves, the AI framework now receives the result the AI calculated and incorporates the statement of the player.

```
player: I kick the rat.
result: The move was very effective.
response: The rat squeaks and flies across the room.
```

Tests have shown, that using numbers and scales to represent the result of the attack evaluation by the rule framework may produce unwanted responses. GPT has been trained on natural language and therefore, using such produces the best results. For this reason, the result of the rule framework is given as plain text. It is important to note, that the player will only see the response of the AI, not the result the rule framework created. This is because the rule framework itself is only capable of producing four states: very ineffective, ineffective, effective and very effective. If the player had to constantly read these sentences, the battle would become dull. Therefore, only the response is shown.

The same goes for the evaluation of the damage. The rule framework calculates the damage and shows it to the AI Framework. It also includes the player's move (as 'attack') and how that move has been evaluated previously (as first sentence in 'results').

```
attack: I swing my sword at the orc.
result: You slice through the arm of the orc. The orc has 2
     of 10 health points.
response: It looks like the orc is in great pain.
```

It is important to give the AI framework the information of how many health points are left, since this statement is meant to communicate an estimation of how close the player is to victory. This way, the AI can learn that numbers closer to the total health points do not require a severe description. During testing, sometimes the AI even generated texts to taunt the player if the enemy was still at a higher level of health points.

With these prompts, a battle move looks like this. First, the player states their move. The rule framework generates its effectiveness based on the character stats and skill checks. Using the player's original input and this evaluation, the first prompt, that describes the results of the player's attack, is generated. At the same time, the rule framework calculates the damage done based on the weapon's strength and another skill check. The damage is set against the defense of the opponent and the remaining hit points are calculated. Now, the second prompt, that describes the condition of the opponent after the attack is generated, as mentioned above.

7.3.3 Entity Extraction

Named Entity Extraction (NEX[16]) is (in the scope of this master thesis) the process of detecting entities uttered in a statement and then classify them into categories that the Rule Framework understands. The rule framework can then search the content for these entities and, if they do not exist, create them thanks to the process of content creation. This, of course, brings the danger of the player going wild and exploiting the AI to create the craziest items. If their utterances would not be questioned and just analyzed, what would stop the player from summoning an eldritch demon that just eats everything. This is where the authority over the shared imagined space comes into play. Most PnP systems lay this power in the hands of the GM. In this system, VGM is given total authority over the shared imagined space. During programming, this means, that only statements uttered by VGM would have to be analyzed for unknown entities. Since statements made by VGM already undergo plenty of restrictions based on the rule framework, it is assumed, that

---

[16] Usually, this process would also be called Named Entity Recognition. However, to not confuse it with the process used by the NLP in the rule framework, the way VGM uses this method is referred to as Named Entity Extraction. In my opinion, this is more fitting, since the AI is not trained to 'recognize' entities (because they have not been shown to it before) but rather just 'guess' which entities could be relevant.

VGM is always correct.
The process of detecting entities within a sentence is a pretty straightforward task. Every word that could be given a class is considered an entity. For example: 'cellar' is a location, 'sword' is an item, 'two' is a number.

```
description: You can see a book lying on the table.
entities: book (item), table (furniture)
```

The extracted entities can then be checked by the rule framework, whether they already exist in the content or even needed, based on how they are being classified.

7.3.4 Content Creation

Another essential part for the AI framework to do is to create content. For one, usually the GM is the one writing the story, so it would only be fitting for VGM to also be able to write content. This is actually one of the main reasons for the content being structured as detailed as described in section 5. This way, the AI can be shown the previously existing content and then create their own stories based on that. This topic, however, requires even more preparation, training supervision and training material than the scope of this master thesis allows.
Creating content also happens on a smaller scope to account for seeding. The AI may, for example, have used a certain item or location in its response generation. While the training aims not to have the AI do that, until now, every so often it still happens. Furthermore, the content may already contain some smaller seeds, like a dinner plate or a glass of water. Usually, when a GM describes such things, they tend to do so to create an atmosphere and draw an imaginary picture of the scene. However, that will not keep the players from using such items. The GM would have to react accordingly. The same is to be expected from VGM. It has to be able to at least detect if some sort of entity like an item or location has been named, that is not already existing in the content or the rule framework. In such a case, it would have to populate such entities with a description or properties, depending on what type of entity it is.
The example prompt for this task looks simple, since it is already prewritten by the author. The only necessary step is to rewrite it in more plain text than the JSON format used by the rule framework. This is

done to reduce potential errors that come from the AI potentially not completely reproducing proper JSON formatting.

```
id: service room
description: A very small storage room. You can see some
    utensils for cleaning. Furthermore you see a fuse box
    mounted to the wall.
```

A similar approach can be used to generate items. However, as mentioned above, since Events and Content Situations have a way greater impact on the whole story presented, they require a more detailed implementation and prompt design. This will be mentioned in section 9.

Part III

Evaluation

8 AI Evaluation

Playtests have shown, that the rule framework was very stable and reliable. It only failed at the points where it was designed to fail, which were situations, the AI framework was supposed to take actions. Afterwards, the AI framework itself has been designed to fill these gaps. However, Given the nature of a PnP like role playing game, testing the success of the AI framework is not as straightforward of a task as testing the rule Framework.

The initial main intent behind this master thesis is to use an artificial intelligence to extend the pre-existing content of a computer game. This was done by trying to adapt a PnP game into a computer game. To evaluate the success of this endeavor, the performance of the AI has to be rated in relation to how it feels to play with it. Just performing more play tests, however, would most likely not yield conclusive results. A player may for example decide to just follow the given path. In such a case, the current state of the AI framework would not do much. A player may also decide to do the opposite and exploit every little problem they find. Although such a behavior would clearly show, *when* the AI framework fails and due to the nature of the system design also give hints on *why* it did, it would most likely not be useful when trying to figure out, how to solve a problem. To test specific parts of VGM this way, a large amount of play testers would be required, to find and target regularities.

Another way to approach the evaluation of the AI framework would be to prime the play testers to ask specific questions. They could be asked to make certain statements or try to move the conversation towards certain situations. This, however, would also mean, that the players can already guess, what statements are made by an AI. Furthermore, given how short the current version of the story is, players would most likely ask similar questions about similar topics. Although the player's input and VGM's outputs could then be evaluated, this would neither be an effective solution, since still a lot of play testers would be needed. Given that the evaluation is about the response quality and consistency, a bigger model than the one mainly used for testing would be needed.

Sadly, my financial and computational resources are too restricted to make use of bigger models in the quantities needed.
The only advantage of conducting full play tests of the entire game would bring, is that asking questions multiple times would give data on which the reliability of the AI could be evaluated. This, however, has already been done during testing and developing of the AI prompts in section 7.2.3. The design of the current prompts is the one that turned out the most reliable after numerous iterations of various formulations and formattings, while at the same time giving the AI enough freedom for creativity. Therefore, evaluating reliability is not a primary task at the current state of VGM.
Given that the scenarios in which the AI framework has to generate a response are pretty well-defined by the rule Framework and given that the Rule Framework has already been tested, a more urgent question is how the quality of the AI is perceived and how it compares to the human written content. After all, having the player notice the difference between human and AI-generated content could break the immersion of the game. With that question in mind, a survey shall be used.
Using this method, the questions can be designed in a way, that all scenarios can be specifically tested. The participants, however, would not be spoiled towards a certain direction. Using this method brings the benefit of attracting a higher quantity of users. At the same time, questions can be designed to still get qualitative responses of users that want to take the time to answer these. This way, more data is generated that can be evaluated and used to continue the work on VGM.

8.1 Questions

To gather some context of prior experience of the participants, they first were asked whether they already played some form of PnP and in which role and if they have already used some sort of AI and in which context. The rest of the survey consists of ten questions, where every question is built the same way. The participants are being shown a short excerpt from the game with a short intro text with relevant information for this scene. These excerpts are scenes that could or already have happened this same way during a game[17]. The scenes consist of three statements. The first statement is based on something, that happened

---

[17] Only one scene was altered slightly to ignore the rule framework for this particular content to test the capabilities of the AI Framework.

during the game. The second statement is the player's response, either as it actually happened during play testing, or it was made up based on what could have been asked during this situation. The third statement is the response generated by VGM based on the decision made by the Natural Language Generator. This means, if it found an answer in the content, this answer has been picked. If the content did not include the correct answer, an AI statement was generated the same way as explained in section 7. These responses have not been cherry-picked. Whatever has been generated by the AI first was used for this survey. For better evaluation and comparison, exactly 50% of the situations have been generated by an AI, the other 50% was found in the content and therefore has been written by a human. The entire survey can be read in attachment 12.4.1

For each of the situations, the participants were asked the same three questions. First, they were asked whether they think the last statement of the scene has been written by an AI or by a human. After that, they were asked to rate the overall perceived quality of that statement. Lastly, they got the optional opportunity to comment on that statement. This way, they could reason, why they picked that particular answer or just in general write, what they think about the whole situation.

The meaning of the first question is first of course to evaluate, how detectable statements generated by an AI are. If they are indeed noticeable, a tendency towards the 'correct' answer would be expected. If they, however, are not detectable, the distribution between the options would tend towards 50%. For this thesis, the later would be the ideal result.

Asking the participants to rate the quality of a certain statement serves multiple purposes. Foremost, if a statement was generated by an AI, answering this question already gives an idea of 'how good' an AI is perceived. This question, however, is also being asked, when the participant faces a human written statement. The reason for this is to not prime the participants to 'know' when they are facing an AI statement, but furthermore to afterwards compare the perceived quality. Giving the participants the option to rate the quality *every* time and not only when they previously stated they think they are facing an AI statement also gives the option to evaluate whether the perceived quality of a statement correlates to the participant's assumption made beforehand. To sum this up, the important steps of evaluating this question are:

1. What is the overall quality in correlation to the perceived generating entity[18]?
2. What is the overall quality in correlation to the actual generating entity?
3. How do the first two compare?

An ideal result for this thesis would be to have an equal distribution of perceived quality regardless of the generating entity. However, if statements actually written by a human would overall be rated worse than statements actually written by an AI this would also be a positive result, since such a result would mean, that the human author writes worse texts than the AI[19].

The last question was to give the participants an option to write, what was on their mind when reading the question. The reason for designing this survey as a simple multiple choice survey was to attract more participants. Given that the main purpose of holding this survey was to evaluate the *perceived* quality of the AI, more people to perceive were needed. However, such a survey is in no way interactive and very restricting. While for the shier purpose of this one task it would absolutely suffice, personal observations have shown that while talking about a game lead by an AI entices people to state their mind about more than just these two questions. By giving the players the option to do exactly that, way more data can be collected about what the participants would expect from the AI or the game in general. These results help to conclude, whether the AI can indeed be used in the context of a computer game. Furthermore, they can be used to design ways on how to improve and further develop VGM. Therefore, for this particular question, there are no expectations whatsoever. It only shall invite the participant to reason, why they picked their answer.

The evaluation will be held in two parts. First, during the quantitative evaluation, the results of the first two questions will be summarized. Secondly, during the qualitative evaluation, a closer look at some of the comments from the third question will be taken. Based on these evaluations, four questions shall be answered:

---

[18] The 'generating entity' refers to the one who wrote a certain statement, the AI or a human.

[19] Since the human author in this case is me, although at first, it would make me sad, I also knew that I could do way better. Story writing and programming at the same time seems to have a bad influence on the writing style.

1. Can the player differentiate between AI and human?
2. Is the quality of the AI-generated texts sufficient to be used in a game?
3. Does it matter for immersion whether the AI or a human wrote the answer?
4. What do players like or expect from VGM?

8.2 Quantitative Evaluation

Overall, a total of 39 people have participated in the survey, seven of which (19%) stated, that they already had prior experience with an AI of some sort. A summary of the quantitative results can be found in attachment 12.4.2

8.2.1 Question 1: Categorization

The first question, whether the participants think, a certain statement has been generated by an AI or was written by a human, has a definitive correct answer. Eight out of ten times this answer has been answered correctly, in all cases by more than 60% of the participants. Furthermore, it can be observed, that on average, every participant has chosen the AI 3.3 out of 5 times correctly. The human was chosen 3.2 out of 5 times correctly. Wrong categorizations, on the other hand, were only made 1.8 times for an AI and 1.7 times for a human.

If the participants only picked at random, the distribution of results would follow a Gaussian distribution. This would mean, that for this number of participants, 98% of the times, 50% plus or minus 15% of the participants would have guessed correctly. The distribution of 'correct answers therefore would be between 35% and 65%. However, this is only the case for 50% of the answers. Meaning, for these 50% it is likely, that the participants did not 'guess'. The other 50% of answers include both of the overall wrongly answered questions (40%). Judging purely by the numbers, it is possible, that these 5 questions were answered at random.

The first scene that was on average wrongly categorized by more than 59% of the participants was the scene one. In this scene, a location was described. Going by the results of this question, it seems, that this description was of similar quality as the participants expected a human

written description to be.
The second scene that is of interest regarding its categorization was number eight. This time, however, the distribution was way closer. Exactly 50% percent of the participants categorized wrong. In this scene, VGM describes, that the player manages to open a hatch set in the floor. The description is very short and merely focuses on the fact, that the player succeeds in their attempts. There are no hints whether a human or an AI has written that statement.

8.2.2 Question 2: Quality Rating

The participants were asked to rate the quality of the statement on a scale of 1 to 5 with 1 being the worst and 5 being the best quality. Overall, every scene, regardless of their generating entity, was rated with a quality of 3.76. Scenes, that were indeed generated by a human, were rated at 3.93, whereas scenes generated by AI were rated on average at 3.58. The same distribution holds up for scenes that were categorized as generated by a human or by an AI, regardless of the actual generating entity.

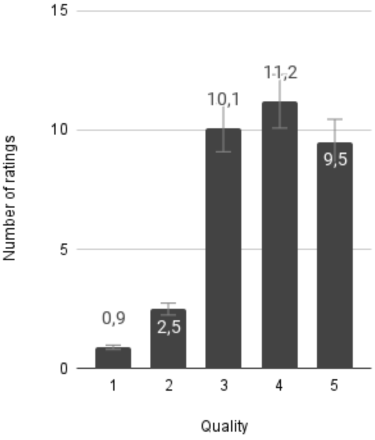
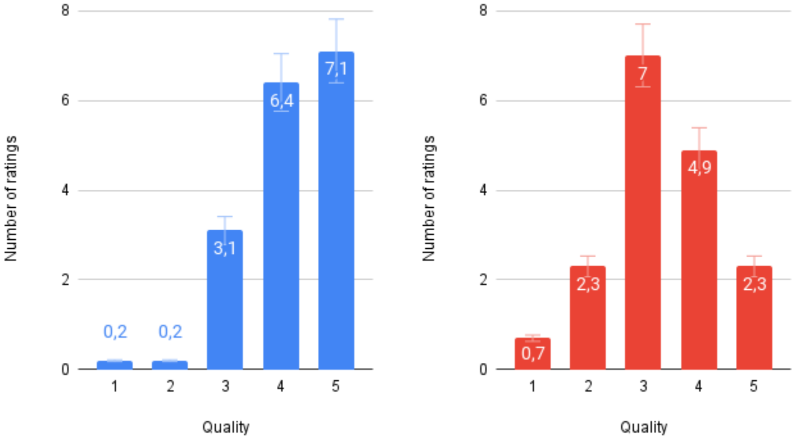


Figure 3: Combined quality rating distribution

At first glance, this leads to the assumption that statements gener-

ated by an AI are on average perceived as worse than statements written by a human. This can mean that either if a statement is categorized as an AI-generated statement, it is perceived as 'bad' *or* if a statement is perceived as 'bad' it is categorized as an AI statement.
Separating the responses by categorization regardless of the generating entity yields interesting results in this question. If a participant categorized a statement to be human written, they rated the quality on average to be 4.12. On the other hand, if a participant categorized a statement to be generated by an AI, on average, they rated the quality to be 3.3.



(a) Quality for statements perceived as human-written

(b) Quality for statements perceived as AI-generated

The most interesting scene in this regard is number nine. Of all the scenes, this one has been rated the worst (avg. quality of 2.71) by people who categorized this scene to be generated by an AI while at the same time being rated the second best (avg. quality of 4.6) by participants who categorized this scene to be written by a human. This scene was, in fact, written by a human.
All in all, it can be observed that participants who categorized a scene to be generated by an AI tended to rate scenes that have *actually* been generated by an AI to be 3.42 on average, while they rated scenes that were actually written by a human on average at 3.23. In other words: although participants who categorized a scene as being generated by an

AI tended to rate this scene lower if written by a human. The scenes that were *actually* generated by an AI averaged to a higher quality by these participants. This observation leads to the assumption that participants tended to categorize a scene as generated by an AI if they perceived the quality as 'not good', and not the other way around.
An attempt of proving this assumption can be made when observing the opposite, meaning the rated quality of texts generated by an AI but categorized to be written by a human. And indeed, this observation can be done with scene four. This scene has been rated the best (avg. 4.73!) of all the participants who categorized this statement to be written by a human, when in fact it was actually written by an AI. This supports the assumption, that the perceived quality of a statement influences the categorization of the generating entity.

8.2.3 Conclusions

Based solely on the quantitative evaluation of the survey, a few careful conclusions can be drawn. The first being that it seems like the participants tended to categorize a statement to be generated an AI because they perceived the quality as worse than the average. Although the examples shown during the evaluation of the second questions are only few in numbers, it is the sheer significance of deviation from the average that supports this conclusion. Since the participants had the opportunity to reason why they picked a certain answer, there is an opportunity to see *why* some participants perceived an answer as 'bad'. A closer look at these statements will be taken later on.
The second conclusion drawn from the evaluation of the first two questions is, that AI already seems to perform better, than the participants expected. The first support for this assumption is the fact that participants who categorized scenes to be generated by an AI regardless of whether they were right or wrong also rated the quality of these scenes higher if they actually *were* generated by an AI.
The second support for this assumption is the evaluation of the scenes four and ten. Both scenes were very close in perceived quality (3.76) to the overall average quality (3.80) and were written by an AI. Scene four was rated with the highest quality (4.73) by people who categorized this scene to be written by a human, leading to the assumption that it was categorized this way because it was perceived as 'good'. Scene ten on the other hand, as mentioned above, revealed an interesting result when

shown to participants with prior experience with AI. 80% of these participants categorized this scene to be written by a human. This group on average rated the quality of this scene at 3.8, which is higher than the average rated quality of this group (3.58).

Of course, these two conclusions are purely based on a few examples The overall tendency towards participants being able to detect AI statements and also generally rate these statements at a worse quality is clear. However, the difference in categorization as well as quality rating in most cases is minimal and indeed blurred by the observations that lead to this conclusion.

The most important conclusion that can be drawn from the evaluation of these two questions, however, is the fact that the average perceived quality of AI statements (3.62) is very close to the average perceived quality of human written statements (3.92). After all, the participants were specifically being asked whether they think a statement was generated by a human or an AI, meaning, they already knew, that *some* statements had to be AI generated[20]. The question remains, why participants perceived certain statements as worse than others. This question shall be answered during the qualitative evaluation of the survey.

8.3 Qualitative Evaluation

For the qualitative evaluation, a closer look at the comments given by the participants shall be taken, to draw further conclusions as to why certain answers have been given. All comments can be read in attachment 12.5

8.3.1 Superimposed Answers

Several times the participants directly stated that they could not decide, if a certain statement was written by an AI or a human. The only reason they picked one of the two was because they had to. Other times participants commented in a way that showed, that if they had not been asked, they would not even have questioned, whether some content was written by a human. One comment describes that very well. In scene seven, VGM states that there are empty boxes in the room. The player then asked what they can see in the boxes[21]. The

---

[20] Although one participant interestingly categorized every single statement to be written by a human.

[21] Even though this may seem like a pointless question, since it has already been stated, that the boxes were empty, players still asked it during play testing.

AI framework (correctly) generated the answer, that the boxes were empty. The participant commented: 'It straightforwardly answers the question. So, there is nothing [wrong] with the answer. But it seems like a very basic sentence, making me think this was an AI answer.'[22]. The answer is contextually correct. This fact alone would not leave any clues whether it is generated by an AI or written by a human. Another aspect of this answer is *how* it was delivered. Considering that it has already been answered previously, this question could be categorized as pointless. Some GMs answer pointless questions very straightforward, letting the following awkward silence ask the player, what else they would have expected. Other GMs may build up some tension, describing for example the suspenseful look into the boxes, just to have the disappointing fact that they still are empty have even more of an impact. Either way, assuming the player has read the previous description of the room and the boxes, they could not expect a different answer. Therefore, the only reason, why they would think it has been answered by an AI is because they knew it was a possibility.

8.3.2 Context

The biggest problem with evaluating the performance and perceived quality of VGM with a survey is, that only small scenes have been shown. This way, a lot of context is lost. Although, short introductions were used to try to make up for this problem, judging by the comments, the answers still were sometimes based on the lack of context. Luckily, some participants did indeed state, if they had a problem with the context, even if they did not know that.

One of the more commonly stated comments were the participant not liking the style of the statement. For example, scene one was commented with 'It's a bit brief and not detailed enough. One sink and a table and no more other stuff in there?'. The participant rated the quality at 4 and correctly categorized this scene to be generated by an AI. The overall pretty good quality rating leads to the assumption that the participant does not dislike the statement itself. Their comment was only meant to reason their categorization. And indeed, the description in question is pretty short. However, what the participant did not know, almost every single location description only consisted of two or three sentences. This was the chosen style of the author, and the AI did manage to copy that style very closely. It is very likely that if

---

[22] Scene: 7; Rated quality: 3; categorization: AI (correct)

the participant had known other location descriptions, they would have changed their answer since it seems like the response was based on the fact, that the answer was short.

Another similar remark for the same scene stated, 'It fits as [a] working, consistent answer [...]. It's plenty enough for a use in a small text-based AI game, but seems to be also limited there.'[23]. Trying to style the entire game as a 'small text-based game' was indeed the reason, why the author chose to shorten the texts. This way, more content could be generated, and play testing times were decreased. At the same time, the AI had less to learn, and its efficiency could be evaluated faster. Therefore, even though the participant was correct in their assumption, this description was generated by an AI, the question remains, whether they had picked the same answer, if they had known the overall style.

Another context related issue arose whenever multiple events happened during one scene. In scene six, the player wanted to go through a door to which they already previously had found a key. The answer given by VGM described, that the player opened the door using the key they found somewhere else. During the real game, this shortening of the scene would only happen, if the player had already tried to open the door and therefore had known, that they needed a key to unlock it. If that had not been the case, the player would have been informed that the door was locked beforehand. One participant pointed out that '[...] it should be noted first, that this door is locked, and you can't get through, unless you have a key.' [24]. Mistakenly, this fact was not mentioned in the introduction of the scene. Therefore, the frustration expressed in the following remark is understandable. ' [...] It feels patronizing not to give me a chance to figure out that the [key] I found fits here. I want to figure that out by myself.'[25]

Assuming, that the negative rating of the statement was given because of the missing information, this and other similar responses do blur the overall perceived quality of the AI and the human author. If this is indeed the case, the previously drawn assumption, that participants would categorize statements they perceive at a lower quality to be generated by an AI, would be further validated.

---

[23] Scene: 1; Rated quality: 3; categorization: AI (correct)

[24] Scene: 6; Rated quality: 3; categorization: Human (correct)

[25] Scene: 6; Rated quality: 2; categorization: AI (not correct)

8.3.3 Style

The last and most significant observation that can be made, if the participant directly commented on the style of the text. Comments like '[...] I picked AI because [I] thought I would have said [something] like 'you pick it up' first. And I would have changed the order of the sentence or use two sentences...'[26] or '[...] I as a [GM] would not give [that] [i]nformation or say it in [concurrence] to the [aforementioned] lack of vision (as far as you can see, there is...[])'[27] show, that the participant's quality rating was influenced by the fact that they personally would have used a different description. Both participants stated, that they already had prior experience with PnP games, one of them as player, the other one both as a player and a GM. Therefore, the comments make some valid points. It is likely, that a human GM would have used a more interesting description during a PnP game[28]. However, both of these scenes were actually written by a human. Since all the human descriptions are pre-written without knowing the *exact* context of how and when the player reads the letter and if they want to pick it up first or such, the author decided to keep the answers as general as possible. In these cases, it is likely, that the participants disliked the style *how* the game is referred.

In quite a number of cases, the participants plainly stated, that they disliked the style of the answer, no matter if they were generated by a human or an AI. A lot of the comments criticizing the quality of a statement of a human were down to the fact that has already been explained before, or simply the style of the author. It also happened a few times, that the participants rated a statement badly because they expected more from the answer, although arguably there was nothing else to say. This can be well observed in scene five. Here, the AI generated a description for a carpet in the corridor. Some participants stated that '[the] design of that carpet sounds not good'[29] or that '[...] A human might have improvised and had a more interesting description.'[30]. While it may be true, that the design could have been more interesting, it would not change the fact that this is how the carpet looks now. As another participant stated: 'It's a very simple answer, but the question

---

[26] Scene: 2; Rated quality: 3; categorization: AI (not correct)

[27] Scene: 3; Rated quality: 2; categorization: AI (not correct)

[28] At least I would have done that.

[29] Scene: 5; Rated quality: 1; categorization: AI (correct)

[30] Scene: 5; Rated quality: 2; categorization: AI (correct)

wasn't too complex either.'[31]. Ultimately, however, the criticism in style is not misplaced. During a longer and more intense game, it is fair to expect more interesting and intriguing answers. However, to keep the complexity and rate of errors low, the AI has particularly been trained to focus on shorter answers. Tests have shown that if the training material was changed to be more complex, the generated texts would also have changed in style, however the longer the texts were, the more likely it was for the AI to go off track or seed certain things as can be seen in scene ten.

The last observation regarding style was made when focusing on scenes in which players expect certain emotions, for example in scenes four and ten. One participant, who correctly categorized scene four to be generated by an AI, stated: '[...] no description on how the player feels or, sound, surprise [...].' They rated the quality at 3. A similar remark can be found in scene ten. The participant also correctly categorized this scene to be generated by an AI at a rated quality of 3 and commented: '[...] not [much] tension in the sentence, just describing. [I] mean, there is someone watching the house, that's spooky or at least surprising (I think).' It can be assumed, that these scenes were perceived as emotionless because no clues were given as to what the player should feel. However, due to personal preference, the author particularly decided to do exactly that. Since VGM is meant to be used for role playing games, the players themselves would have to decide, whether their character is for example scared, frustrated or even bothered by something they observe. This, of course, leaves players on the side that expect a more lyric approach, where the storyteller at least suggests feelings either directly by saying them or indirectly by having the player shiver or get cold.

8.4 Results

Thanks to the conclusions drawn in sections 8.2 and 8.3 the overall questions of this survey can be answered

8.4.1 Can the player differentiate between AI and human?

The short answer is yes. Judging only by the numbers, the player using VGM could on average be able to tell whether a statement was generated by an AI or pre-written by a human. However, in 50% of the

---

[31] Scene: 5; Rated quality: 4; categorization: AI (correct)

scenes, the answers are categorized very close to a random distribution. Taking other conclusions from the qualitative evaluation into account, this answer is blurred even more. Some participants stated, they only categorized a statement to be generated by an AI because they *knew* it was an option. Some stated, it was indistinguishable and therefore guessed correctly only by chance. Other times, they categorized correctly, but judging by the comments, they did so because they disliked the quality or style of a certain sentence.

8.4.2 Is the quality of the AI-generated texts sufficient to be used in a game?

Going by the numbers, the quality of the AI-generated statements is very close to the quality of the content written by a human. However, taking comments into account, this answer is again a bit more complicated. Quite a few times, participants rated a statement at a lower quality because they disliked the style. The AI, however, just adopted the writing style the author used. This would lead to the assumption, that these participants would have rated a human statement on the same topic, for example a description of a location, equally bad. Based on these observations, the average quality rating of both the AI and the human would get even closer than the numbers suggest.
In my opinion, for the context of VGM, that close of a result between human and AI is very positive. Since the AI learns from the author, the entire game can only be as good as the content the author writes. Therefore, in the context of this thesis and the current state of VGM, this question, whether the AI quality is sufficient, can be answered with yes.

8.4.3 Does it matter for immersion whether the AI or a human wrote the answer?

This question is essential for VGM, since the system uses both AI-generated statements and human written content. The immersion would break easily, if the player noticed a big difference between the two. Answering this question in more detail, however, requires way more play testing and comparison, especially between players who know they could be talking to an AI and players who don't know this. That statement especially holds true considering that immersion is not only based on the quality of the texts, but also context and how the AI reacts to it.

However, assumptions can be made using the previous two questions and the comments on some of the scenes. Given that it already does not seem too easy to even detect the AI and the perceived quality of such a statement is already pretty close to the overall average, it is likely, that players would not notice an overall difference between AI-generated and human-written content. This assumption is backed by observations found during the quantitative evaluation. Therefore, it is fair to assume, that players would not really notice a difference between the two types of statements. Immersion would most likely not break.

8.4.4 What do players like or expect from VGM?

Until this point, during evaluation mostly more negative comments were observed to understand, what factors might have influenced the answers. Now, to answer the given questions, a few of the more positive answers shall be considered, to understand, what exactly made the players like a certain statement. While evaluating the comments, it mostly does not matter, whether the original statement was written by a human or generated by an AI. As mentioned before, it is possible for the AI to imitate certain things from the training material. Therefore, if there are positive factors in a human statement, they can be implemented in the training material to have the AI copy such behavior. If the participants liked something in a statement generated by an AI, the training material can further be written in such a way. Of course, it is impossible to train the AI in a way, that it generates text to the liking of everyone[32] the same way it is not possible for a human to write a book that is liked by everyone. Doing so is not even the point of this evaluation. Since it is already clear, that the style of the AI can be changed, changing the training material written by an author could easily change the target group. The only reason to answer this question is to increase the perceived quality of the AI. Furthermore, some behavior may already be liked by the participants and can therefore be further implemented in the training material used in other cases.

Rules are often the least interesting part of a pen and paper game. Many GMs therefore decide to use a lot of flavor to hide the rules. Dur-

---

[32] Although the AI could in theory trained to learn what writing style the player preferred and adjust the text generation accordingly, but this is a topic for a new thesis.

ing development of the framework and training of the AI, the decision was made to never actually directly state any rule relevant decisions. Whenever something was decided by chance and rules were applied to determine an outcome, only that outcome was shown. As the author wrote the content, flavor text was used to hint, that rules and roles have been applied, giving the player a sense of success. Scenes six and eight were examples of such situations. Participants stated that they '[...] like that the check is kept secret.'[33] or that they find the hidden role and flavor '[...] very suspenseful'[34]. The same reactions were observed in scene four. One participant directly stated: 'I like that [it is] not just damage numbers. I would love if this would be kept up when an attack actually hits, too.'[35]. In fact, even though these scenes received positive and negative feedback in similar amounts, judging by the comments, there were no complaints about the fact, that the roles were hidden.

Flavor helps a lot with setting the mood and suggesting certain emotions without directly stating them. Scene seven received some comments that stated, the participants would have wished for more flavor text. 'The answer could be more varied, since the boxes being empty has been stated already.'[36]. '[...] I would have liked "they are empty" better because I know that the answer is about boxes. The word "boxes" doesn't need to be repeated.'[37]. This definitely is something, the training material given to the AI could improve on. For now, the AI was trained to give short answers and the focus of its training was put on the fact, that they are as 'correct' as possible. With a better understanding of what's important, however, the training material can be shifted, to use a lot more flavor.

Participants often picked up on certain words to add a bit of flavor. This was an unintentional side effect of the style of the author. He tended to use one or the other uncommon word during writing, and the AI picked up on that. In scene one, the AI used the word 'rickety' to describe the table in the room. One participant stated '[...] Tough, I rarely ever heard that word "rickety"'. They liked the answer with a rated quality of 5. However, training the AI to use as many uncommon

---

[33] Scene: 6; Rated quality: 3; categorization: Human (correct)

[34] Scene: 6; Rated quality: 5; categorization: AI (not correct)

[35] Scene: 4; Rated quality: 5; categorization: Human (not correct)

[36] Scene: 7; Rated quality: 3; categorization: AI (correct)

[37] Scene: 7; Rated quality: 2; categorization: AI (correct)

words could lead to the players becoming suspicious. This particular participant categorized this statement to be generated by an AI. They may have done that because of the uncommon word, since they directly stated it. However, participants categorizing a statement to be written by a human *because* of uncommon words can also be observed. Another participant who commented on scene nine stated: ’because of the tension the sentence creates, [I] could image someone telling that. ghostly play - [I] think AI would have chosen another adjective. [...]’. They correctly categorized this statement to be written by a human and rated it at a quality of 5. Having the AI pick up on such a style will most likely have a positive influence on the perceived flavor.

Seeding is probably the most challenging task for VGM, but seems to resonate very well with how the players may perceive the quality of VGM. Scene one was generated by the AI to evaluate, how well it creates locations. Overall, this scene was rated at a higher quality (3.87) than the average (3.76). Most participants categorized this statement to be written by a human and commented: ’[...] The last sentence gives a sense of mystery and suggest a way [how] to drive the story [forward].’ with a quality rating of 5 or ’[...] it does include a very specific information about the smell of the kitchen. This implies that the Player should want to investigate that information.’ with a quality rating of 4. The same can be said for scene ten. Many participants liked the spooky atmosphere and the fact, that a new plot point was seeded. However, one comment described the actual problem perfectly. ’I’ll go with human because even though this is perfectly possible for an AI, introducing comprehensive new plot points might be a challenge’[38].

It is absolutely correct, that implementing new seeds would be a problem. However, that does not stop the AI from generating them. Once the rule framework decides, a statement has to be generated by the AI, there is no further supervision as to *what* it generates. Tests have shown, that the rule framework can pick up on certain seeds it can categorize like locations and items. Implementing new categories like NPCs, weapons or creatures is absolutely possible as well. Further tests, like the generation of a location description in scene one, have shown, that once a new seed was categorized, the AI can generate the content for it. However, some new seeds do indeed propose a bigger challenge.

---

[38] Scene: 10; Rated quality: 3; categorization: Human (not correct)

While further investigation towards the smell of the kitchen in scene one could most likely easily be improvised by the AI framework (since this could be explained with short descriptions like rotten food) more comprehensive seeds like an unknown figure in the woods are definitely more challenging. In short, the current state of VGM proves, that the framework can follow through on seeds, but developing more complex storylines behind them needs a finer implementation.

Part IV
Conclusion

9 Further Work

9.1 Scenario Detection

VGM is a system that uses numerous rules to tell an AI how to react to user statements in various scenarios. Some of these scenarios have already been identified. The rule framework can detect these scenarios and collect the necessary information the AI needs to properly respond to the player. In the current stage of VGM, these scenarios are: Response to unclassifiable statements regarding content description, detection of unknown content, generation of classifiable content and battle situations. Play tests of VGM as well as common PnP games, however, have shown, that there are way more situations that may occur that need the creative work of an AI. To find such situations and develop rules that help classify them to improve the generated texts of the AI, VGM now needs a lot of play testing. Ideally, another story would be developed that provokes these kinds of currently unpredicted behavior. Afterwards, rules and regulations can be extracted and implemented.

9.2 Rephrase Content

Ultimately, VGM is supposed to work in such a way, that every text is generated by an AI after the rule framework has specified the circumstances. But not only texts for specified scenarios have to be written. Having the AI re-write the content allows having the player's input to be implemented in the response, making them feel like VGM actually reacts to what the player says instead of just reading a book. This would mean, for example, that if the player wanted to take a look at the note from scene two of the survey, VGM may as well have added, that the main character first walks over and picks it up.

Furthermore, texts, that were already implemented in the content, may have to be re-written as well. Currently, if the AI generates a text that changes something in a scene because the player wanted to, that exact change would be mentioned and not be forgotten, but it would be mentioned in a separate sentence. For example, if the content described an empty bathtub and the player were to fill this bathtub, the AI would correctly respond with: 'The bathtub is now filled'. However, the description of VGM would not have entirely changed because the previous description is still saved in the content as well. Therefore, the rule framework would first describe the scene as it was before and then add, that the bathtub was filled. Such scenarios would be taken care of by only having the AI generate texts the human reads. It would therefore not only be contextually correct, but also read better.

All these steps have to be taken over a long period of time, most likely the entire development of VGM because they require a lot of testing and generalization to be of a good quality.

9.3 NPCs

During the development up to this point of the state of VGM, some tests have already been conducted to develop a rule set to have the AI impersonate characters. However, this posed plenty of different problems. First, talking to an NPC in a PnP game can have a variety of reasons. Sometimes, the player only wants to have general information about 'what's going on'. Other times, they want very specific help on a certain task, maybe they need access to a room or need a rare item. This alone makes it complicated to define what a certain NPC can or cannot know.

The next factor of uncertainty is always the player. If they have all the options they can imagine, it is almost impossible for the author or developer to consider all the paths a conversation may take. Since NPCs would have to react to the players not only based on what they know, but also how the main character talked to them, especially when it comes to helping them, it would be more feasible to implement a system for character motivation. This, however, can become quite complex, since the motivation may not only change because of the conversation, but also based on the actions of the player and other NPCs. Furthermore, motivation is not just one single number. Different factors can motivate characters in different ways.

One possible way of tackling this particular issue is to first develop a general NPC behavior and directly or indirectly tell the player what to expect. This way, they are guided in a certain direction when talking to the NPC, allowing for a more granular approach of developing them, instead of programming the complete behavior all at once. This may help to find further rules by which more intelligent behavior could be

developed, and the AI be trained for.
The more complex the behavior of NPCs can be created, the more in depth the stories for VGM can be written. The content would not have to be restricted to smaller solo adventures that only require keen observation and logic combination. If NPCs can be used in VGM, the scope of the stories is limitless. NPCs can be used to be the companion of the player, they can keep secrets, they can tell lies, they can develop diplomacy, they can start wars.

10 Conclusion for VGM

Judging by the survey evaluation and expandability of the system, VGM is at a point from which it can easily be developed to a state in which it may take the role of a GM for a role playing game. The entire process, however, may still take a longer period of time since a lot of training and further evaluation is needed for the system to be able to detect all the tasks and generate fitting answers. The key element of this system is to use the rule framework as a long-term memory for the general short-term context of an AI. The more rules can be derived from play testing, the more likely it is to find overlying scenarios to be extracted and passed to the AI framework. Step by step, more and more story relevant content could be created by the AI framework and implemented in the rule Framework. But just because the AI creates parts of the story, does not mean, that the author is out of the game. During a common PnP game, authors can see their story be developed by the players and take inspiration from them. They may even entirely change the direction of what is being told. VGM however opens an entirely new opportunity. It is very well possible for a human to continue the work of the AI the same way the AI did with the work of the human. Therefore, authors are not only restricted to telling their story and watch it unfold, they can also become part of the story and play it for a bit, guided by the AI and maybe draw inspiration from that. Then, whenever they choose to, they can continue writing the story themselves.
Using the rule framework as the long-term memory solves the problem of the AI 'forgetting' crucial information. Judging by the evaluation of the survey, many other problems seem to be 'commonly known' with AI, like bad grammar lack of context. However, recent developments have proven, that the AI is already far more developed than most people expect. Comments of participants of the survey often stated that they

were surprised, if a certain answer had been generated by an AI, when it, in fact, was. Another comment, on the other hand, stated: 'Is this even grammatically correct? It sounds off to me.' and categorized the statement to be generated by an AI. The scene this comment referred to however was written by a human. During testing, I have never read an AI statement that contained 'grammatically questionable content'. On contrary, the AI even tended to correct grammar related mistakes the author made when generating the training material. At least during the development of VGM, I was more likely to make grammar mistakes than the AI.

The other observation that leads to the assumption, that AI is already way better than some people may think, is one simple fact. If only the results of the survey are evaluated that categorized a statement to be generated by an AI and only these quality ratings are being used, the overall perceived quality of the AI is much better than the overall perceived quality of the human writing. In other words: if the participants thought, a statement was generated by an AI they tended to rate it at a higher quality if it *actually was* generated by it, than when it was written by a human.

In my opinion, the main deciding factor whether a human can detect an AI is whether they even know if it is an option. Based on the evaluation of the survey, I think an AI would be well capable of generating interesting and 'nice to read' stories and therefore referee and expand an RPG game.

11 Conclusion for AI

After thoroughly designing, testing and evaluating VGM, a final word can be passed on the initial question on whether an AI can be used within a computer game to complete and extend the content intended by the author. If done for the right tasks, it is absolutely possible. The most important factor in implementing an AI in a game is to know, *when* to use an AI and when to exactly *not* use it. Restricting the usage of an AI and prepare the actual use of it using regular algorithms allows generating way higher quality content, then letting the AI work itself out. With such a strategy, it would even be imaginable to use an AI in other context than pen and paper replicas. It could generate side quests or tell shorter story lines based on the player's interest.

However, not only for generating story content, the use of an AI would be interesting. Instead, it could also be used for systems like crafting. In many video games, simple tasks like filling a glass with water and using this glass of water to extinguish fire are not implemented, simply because it does not hold any value for the actual game mechanic and would therefore be overly complex. Using methods described in the VGM system, however, could make that possible. The AI could then further be trained to use the visual assets to create a fitting virtual representation. All that needs to be done is find a common ground of communication between AI framework and rule Framework.

This also would result in players using a lot more of their imagination, instead of just trying to find the one solution the developers originally placed for them in the game to find. Using this method, computer games could create a way more immersive world that allowed the player to *actually* take a role and not have one superimposed by showing them two options and have them choose one. The players could develop their own way. The AI framework would make their way possible, while the rule Framework keeps track of their changes. In the end, they would either find the originally intended solution or create their completely own end. Either way, the player would have played *their* way and not one of the two the developer decided to pick.

Of course, this does not render the developers and writers job obsolete. In the same way, many modern computer games use algorithms for procedural world generation, the AI would be a tool that the developers could use to expand the functionality of the game.

12 Attachments

12.1 aidungeon

The following is a complete play through of one of the stories provided in aidungeon. Statements marked with a → are my inputs. The algorithm changed words like ’I’ and ’my’ to ’You’ and ’yours’. Other than that, all texts are unchanged.

Kedar is a world of dragons, demons, and monsters. These dark creatures constantly war for control of the land, leaving the few human kingdoms in the shadow of a dark and powerful force. But now, the kingdoms of man are fighting back, slowly assembling an army to reclaim their world.
You are Jason, a half-demon trapper and a member of an anti-demon organization that seeks to rid the world of all demons and things associated with demons. You are currently in Caldion where you are working with the local government to hunt down trappers who are dealing in demon parts. You have come to this city because you have heard that the government has a plan to deal with the growing anti-demon faction. A group of demons are to be sent in to scare them away.

→ You think to yourself, that this is like the dumbest plan ever and try to get an audience with the government.

Approaching the guard at the city gates you state that you wish to speak with the king and give him your reasons. You aren’t sure if this will work, but it is worth a try.

→ You say ”Good Evening. I have urgent business with the government.”

The guard, a human with several piercings looks you up and down and snorts. ”Well, it’s about time someone with a pair of braincells turned up. What do you want?”

→ You say ”I have heard, the government seeks help, so here I am, offering what I can.”

The guard snorts again and strides forward, he looks like he is sniffing you out. ”How much help?” ”What do you mean?”

→ You say ”Stop sniffing. I just bathed in deadly spores and you might get infected. Its deadly, trust me. The help i offer? Well I know my way around the demon and the human side, do i need say more?”

The guard pauses and gives you a closer look, his brow furrowing. ”Demon. You have demon in you. Not even human, but half-demon.” You shift under his gaze, uncomfortable.

→ You say ”Wow, you are pretty perceptive”

The guard looks at you again, this time with a scowl ”You planning on seducing me or something?” You are caught off guard and start to laugh, the guard follows your lead.

→ You say ”Well, if it works, I might just try that.”

The guard grins and lets you inside. ”Just don’t try to touch me.” ”Deal.” you say. You find yourself in a dimly lit chamber with several guards.

→ You take a look around, trying to find out, where to go to talk to higher positions.

You find yourself facing a table with several men in armor arguing. One of them is speaking on a walkie-talkie.

→ You say ”Hello, my dear gentleman. I was asked here to offer my help, so here i am”

”I told you, we don’t need anymore!” one of the men replies. ”Look it will be fine as long as we stay in the safe room!” another says. ”The safe room is not good enough! The demons will probably find a way to get in there!” ”Hey, gentleman” you begin, hoping to get their attention.

→ You say ”Of course the demons will find a way in the safe room. I mean, I did as well.”

”What is it?” ”I think the Kedarnites have made their move, and we need to defend ourselves.” The men pause and think for a moment.

→ You say ”Of course you need to defend yourself, but not by just hiding in a save room.” Your plan is met with mixed reactions. ”We’re not abandoning our allies.” One of them says. ”Our job is to serve the Emperor, and if that means dying for him, so be it.”

→ You say ”And you try to do that by hiding?”

They ignore your question and one of them waves for you to follow him. You follow the man through several corridors until you come across a massive steel door, which he unlocks with a key, and motions you in.

→ You do step in.

You step into the room and look around. The room is dimly illuminated and several men stand around discussing strategies. You recognize one of them as Mike, a captain in the 22nd Regiment.

→ You say ”Hi Mike, what’s up, long time no see.”

”H-hello there! Ah, I see you managed to survive the cuties downstairs” he startles a bit at your presence.

→ You say ”Yeah, they are quite something, am i right? ”

”Ah, hehe, yeah!” He thinks for a moment and fumbles with words. ”G-g-guess so...” ”By the Emperor!” You chuckle and lean against the wall.

→ You say ”Are you alright, Mike? you seem distressed.”

He nods and breathes slowly, collecting his words.

→ You say ”You can tell me”

”Ah, well, Kedarn... it’s been attacked.” You furrow your brow.

→ You say ”Oh has it? when?”

”Ah, well, two days, maybe three.” ”Strange. I thought the Kedarnites had the whole place under control.”

→ You say ”Could it be an inside job?”

”Hah! Don’t be ridiculous, us trappers aren’t that... Anyway, it doesn’t matter. What matters is we’re fighting the second wave now, and I fear it will overwhelm us.”

→ You will have a close look at mike. He does not seem very honest here.

You thank him for the information and turn around. In front of you is a man wearing a black cloak, holding a silver sword.

→ You say ”and who are you”

He responds in a strange language that you don’t understand, but you respond in kind and raise your hands.

→ You say ”So, what’s the plan here”

He replies in kind and raises his hands. ”It is an honor to meet you, human.” ”You’re not much for small talk are you?” you say. ”Small talk is overrated.”

→ You say ”Yeah, you are totally right”

He smiles and nods. ”So, what can I do for you?” ”Well it’s like this....”

→ You want to help you guys to work on a plan to keep the next attack away, maybe learn something about the actions of the demons. If you work cleverly you might just be able to not only defend your city, but also send them back to where they came.

He pauses and thinks carefully before replying. ”Well... We could use some help, but I don’t think the head of the Nigh creatures will be very helpful.”

→ You the Nigh creatures?

He stares at you before replying. ”The Necromancer’s Army of Kedarn, the Nigh creatures are what we call all the undead.”

→ You say ”I see. Well as i said, I think we might be more successful if we avoid a direct confrontation and try to eliminate the problem at its root. We should send out scouts to find out more about the whereabouts of the demons”

He thinks again and chews on a knuckle before replying. ”Ok, but I’m not sure if we should reveal our hand so early.”

→ You say ”That goes without saying, we should stay hidden and keep it a secret”

He nods and looks out over the city. ”I agree, we must not let them see us or they will try to destroy us from within before we can strike out at them.”

→ You say ”So do you think of infiltrating them?”

He thinks for a while before replying. ”I have heard stories of the demonkin, but I have never seen one.”

→ You ”me neither” you lie.

He thinks for a while and then opens the flap on his tent and stands up. Putting his hand out to you, you shake it and he leads you over to the edge of camp.

→ You follow him.

”We should keep our meeting out of sight” he thinks. ”This tent seems suitable.” You nod and the both of you slip inside, you sit on one side and he on the other.

→ You say ”So, do you know anything about Mike? can we trust him?”

”Not sure, but it seems the safest option. He is in contact with travellers across the realms and we can get information quicker through him than any of our other contacts.”

→ You say ”Thats good, we might use that to gather information of the surroundings. Maybe some strange sightings as well”

Your words seem to make him worried. ”I thought you said you had a contact in this town? description, features and all that?” ”I do, but he’s...”

→ You say ”He is pretty busy with work and stuff”

He raises an eyebrow as if disbelieving your excuses. ”I trust him more after spending time with him, but we need him if we are going to save the world!”

→ You say ”So, you don’t trust me enough?”

”Not at this time, but I’m sure you’ll convince me.” You nod. ”I understand, but I’ll find a way, I always do.” But your grandfather seems skeptical.

→ You say ”how are you suddenly my grandfather?”

He is silent for a while as you await his reply. ”I don’t know, but it feels right doesn't it?" You nod slowly, deep in thought.

→ You say "Ok, when is the attack to be expected?"

"In a week's time, the emissaries said." He looks out through the tent opening, seeming to ignore you.

→ You say "Whats going on out therE?"

"The ritual to open a portal to Kedar has failed, the magic that was to speed up the process is gone and so is many of the emissaries. We have to save as many people as we can."

→ You say "Well then lets go and save the hell out of them"

"I'm glad you said that, come on let's go save the day!" Your grandfather smiles and puts his arm around you. "This is why I love you." He says.

→ You say "ok, less hugging, more fighting"

He laughs and you both grab your weapons to prepare for the assault on the demon stronghold. You stand at the head of your army waiting for the attack order.

→ You just dont wait for the order, yell out "CHARGE" and run for it.

You charge at the demons, who give a yell and try to stop you.

→ You tHERE AINT NO STOPPING ME.

You go to swing and one of the demons is severed in two by your sword swing. You swing again and get cut three times but ignore the pain and swing for the master-beyond-the-gate-thing.

→ You cast fireball.

You summon a fireball and let it fly at the thing. It engulfs the creature in fire and you hear a yell from one of your soldiers to your right.

→ You say "suck it up, i am working here"

You swing once more at the emissars but they block the last swing, drawing their swords. "Looks like you got the upper hand now stop me."

→ You say "Oh believe me, I will" and pierce out his eyes with your middle finger.

He screams in pain as blood spurts from him and stinks of blood. "Go, and tell your master that you failed because I pierced your pretty heads too quick." Your troops charge into the demons.

→ You kILL THEM ALL.

You kill the demons in front of you, your troops do the same. But it's obvious that the enemy is not just a mob of demons.

→ You throw your arms in the air, chanting out a spell to summon a gold dragon. When he arrives, you mount it, and command it to set fire to everything. You begin chanting as a red dragon, breathing fire at the demons. You weave some fire magic to increase it's power, causing the night to be lit up by the inferno.

→ You mount the dragon and fly over the battlefield.

You quickly leap on to the back of the red dragon. You raise your hands and command it to breathe fire over the area. The demons are engulfed in flames which they scream out in pain from.

→ You command the dragon to eat up the deamons. As he starts doing so, you yourself unmount the dragon and start eating the demons.

You start eating a demon, taking a bite of flesh and chewing. The blood flows down your chin as you eat another one. After about a minute, the dragon is done and walks back over to you. "You did well sir" He says. "Now mount up. We will arrive at our destination soon."

→ You take another bite of the demon and mount back onto the dragon.

You mount the dragon, which nods at you. "Gruesome" He says in a deep voice, as you raise your sword to start slaying the demons. "Stop!" The dragon shouts.

→ You say "why shall i suddenly stop now? we just murdered a whole Army of demons, one more or less doesn't matter, does it?"

The dragon snorts. "I don't believe you. I think you are afraid."

"No, I just... TRY and stop me, you cowardly cur!" You shout. He laughs.

→ You say "Afraid of what?" and take another bite of the demon.

The dragon looks at you, raises his head and lets out a roar. A roar so loud that it shakes your bones, and even the ground beneath you.

→ You say "Is that all you got?"

The dragon roars, and you feel you spine bending in half. "I'm gonna eat your soul." He roars, and charges at you, moving too fast to be seen.

→ You banish the dragon to the underworld.

You release a wave of negative energy, and the dragon is ripped from the air and sent spiraling to the ground below. You feel your bones reforming and repairing themselves, and you rise up onto your feet.

→ You grow a pair of wings, start rising over the field of burning, dead corpses. you look around, trying to assess how many humans have fallen.

You grow two pairs of leathery wings from your back, and begin flapping them. You rise up into the air, quickly becoming airborne as you are carried by the wind.

→ You stop caring about the rest of your humans and your grandfather. Instead you turn into a dragon.

You turn into a dragon, and find yourself surrounded by a hoard of demons. A large group of violent demons, who love to fight. You find a large group of violent demons, who love to fight.

→ You continue eating the demons, since they taste so amazing.

You fly around in the sky, eating the demons. They taste great! You fly around the area, before one of your demonic brethren sees you. "Ah! Ah, a human!"

→ You say "I aint no human, you fool, how dare you insult me you tiny creature of the land"

You shake your head, annoyed at the demon's disregard for your feelings. ”Demon, I am a dragon, a being of immense power.”

→ You say ”You demons shall flee from these lands, never to come back again. If you come back, i will hunt you, your children and your childrens children through day and night, reality and dream, life and death.”

The demon shakes in terror, knowing you will fulfill your threat. ”I have decided, my brethren and sisters. We will push the forces of evil back once and for all. We will take back what is ours.” Your brethren and sisters agree, and raise you up. ”We will purge the world of the forces of darkness.” The demons flee, never wanting to fight you or your hordes of demonic reinforcements. You raise a massive army of demonic hordes, and flying beasts!

→ You say ”From now on and for all time to come, I shall reign these lands. Every soul that disobeys me will be banished into the underworld. A new era has begun. THE ERA OF ME”

The hordes begin to pour into the world, ready to conquer everything within their path. You rule for a long time, after which you find that your powers are fading.

→ You will turn yourself into an even mightier dragon.

You grow several more sets of wings, each as powerful as the one before. Your powers increase once more and you reign as a powerful dragon, watching over your lands and allowing no one to challenge you.

12.2 Risas Secret

12.2.1 Background story

The player lives in a small village that survives on its own. Farming, animal husbandry. It exports wool and clothing. This way it gets money and pay for tools. The main character is part of a farming family that works on agriculture.
Risa is also living in this village. She is an old, cranky woman, shaman by trait. She never has really drawn any attention to herself and served the village as a healer. But lately, she secludes herself more and more. She is being called by a demon, which forces her to provide her a host body in the human world. For this, she tries to use a wolf for this. Once this is done, the demon can hunt for a better body himself. Risa intends to lead him to the next bigger city.
The story begins with the player seeing a purple light from the well in the center of the village. The glowing comes from one of Risas rituals. She tries to summon the demon in an underground cellar. However, she does not know, that the cellar is connected to the well.

12.2.2 Game storyline rundown

First, the player is told the background story. Where are they, what do they do. Other than that, the game jumps right into the actual story. As the player is headed home from their work, they see the purple light. They can choose to investigate or ignore it. If they, however, seek for help, they will be rejected by the people living in the town, I mean, it's the middle of the night, no one wants to hear about wired lights. This will have an influence on how the people react to the player the next day.
In the second part, the player forms their path. They either go right exploring the dungeon if they are already in the well, or may go explore a bit further. Eventually, they could end up seeking help by the villagers or find decide to go to Risas hut, either because they want to question her or are just curious. In due time, they are supposed to end up in the Underground dungeon.
Here they will face the boss fight, The wolf, that was possessed by the demon in the first night. If the player succeeds in defeating the wolf, they win. If they don't, the village gets destroyed.

12.2.3 Game storyline flowchart

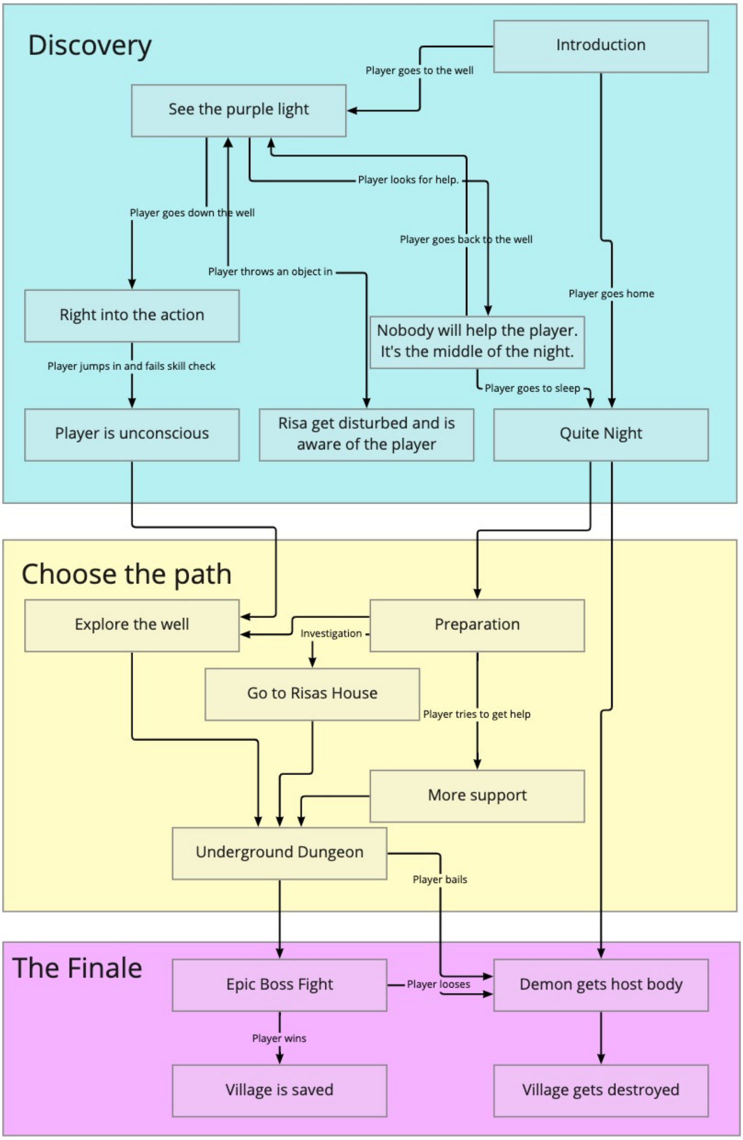


Figure 5: Flowchart of the Risas Secret game story

12.3 Haunted House

12.3.1 Background story

Marc and his wife Rosa recently moved in at the house near the city in the woods. They were a bit wired. They never really made any contact with the city folks and just lived alone. After a while, they suddenly seem to have disappeared.
It so happened that both of them were researchers, trying to find a way to immortality. Marc did indeed find a way, but what he did not know is that he had to pay one life with another. Death took the life of his wife. Marc, however, managed to keep her material form (as ectoplasm) in the human world. This way he accidentally created a medium to the world of death, which can open a portal to this realm He entered this realm trying to get back the soul of his wife. Since he is not dead, however, his subconscious soul has not left the realm of the living yet. It still remains in the house, haunting it, waiting for the body to come back from the realm of the death.
However, No one in the city knows about any of this. They just want to resell the house.

12.3.2 Game storyline rundown

The player is briefly introduced to why they are here and what they are supposed to do. They then are free to explore the house and find what they are looking for. Everything comes down to the player having to find the secret room where the ectoplasm is located. They can do so either by entering the room through the secret hatch located in the fire place in the living room or directly through the front door that is locked by a combination lock of levers. The combination of levers to pull can either be found by keen observation, at random, or by finding a note in the study. The study itself is locked, and can either be opened with the key located in the dining room or by lock picking.
Throughout the entire game

12.3.3 Game storyline flowchart

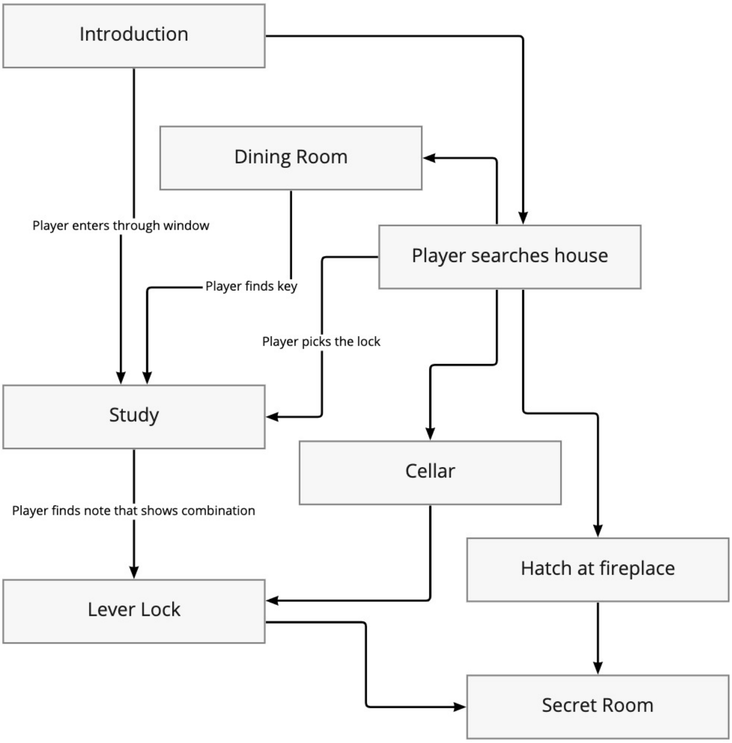


Figure 6: Flowchart of the Haunted House game story

12.4 Survey

12.4.1 Scenes

The following are the scenes, that were shown to the participants. They were always asked to categorize and rate the quality of the very last statement done by VGM. The footnotes were added afterwards and only for this publication. The participants could not see them.
The entire survey was written in a way that made it feel like the participants were addressed directly. For this reason, the introductory texts are written very short, as if someone talked to the participant.

Hello there. Thanks for taking the time to take part in this survey. My name is Benjamin Feder, and I am currently writing my master thesis in system design. This survey is needed to evaluate a certain part of this thesis.
About the thesis: Virtual Game Master (VGM) is a software framework that uses several neural networks to create a Pen and Paper Role-playing experience on a computer. This means, VGM tells a story, and the player can live through it in their own way, by writing whatever they want. VGM then tries to react to the player's intent in a way a human game master would do. The main part of this thesis is to create a system and train AIs to achieve that goal and, based on these results, develop a prototype as proof of concept. The prototype will be in the form of a text-based game.
To put it less confusing: VGM is an artificial intelligence (AI) that is supposed to play Dungeons and Dragons.

About this survey: VGM does not exclusively use AI-generated texts (for now). Some texts are pre-written by an author to give the AI a guideline, what the story and the world is all about. This survey aims to evaluate the quality of what the AI says compared to the texts, that have already been written. In other words, it tries to answer these questions: "Would the player know, when they are talking to an AI?" and "Are the AI responses relevant and interesting?"
How does it work:

1. For each question, first, you will be shown a short excerpt from the game.

2. You will then be asked whether a certain statement from that excerpt was written by a human or an AI. (Don't worry, it does not matter if your guess is wrong, just go with your guts).
3. And finally, you will be asked, to rate the quality of the statement (regardless of whether it has been generated by an AI or written by a human). Was the question answered? Is the answer consistent, or does it contradict a previous statement? Does the statement make sense?
4. Every question will also have a text box in which you can tell me, why you picked that answer, if you'd like to.

That's it. The survey should take about 10 minutes.
One last thing: No personal information will be stored. If you want to be informed about the results of the survey, you will be asked for your E-Mail address at the end of the survey.

At this point, the participants were asked whether they had prior experience with Pen and Paper games, and whether they were Game Master, player or spectator. They then were asked whether they had prior experience with artificial intelligence of any kind. They were allowed to specify these interactions in free form text.

Now, the participants were shown the scenes. They were always asked to answer these questions:

1. Do you think, the last statement by VGM (marked in red[39]) was written by a human or an AI?
2. How do you rate the answer? [40]
3. If you'd like to, you can explain why you picked that answer. (optional)

1/10 [41]
Let's start where all stories begin. With the introduction.

---

[39] In the survey, the last statement was marked in red, to clearly show the statement in question.

[40] The participants had the option to rate on a scale from 1 to 5 with 1 being 'It's terrible' and 5 being 'I absolutely love it'

[41] Note, This scene has been modified for the survey. The house does not have a kitchen, so VGM would say that. However, this question was used to test the content creation capabilities of the AI.

VGM: Far from the city there is a small cabin in the woods. The owner of which has disappeared quite some time ago. Everyone in the city feels uncomfortable around it. As a private detective, you got the job to take a look and maybe find out, what happened here. Perhaps you can gather clues about the whereabouts of the previous owner. You are now in the cabin.
Player: I go to the kitchen.
VGM: The kitchen is small and cramped. A dirty sink is in one corner and a small, rickety table is in the center of the room. There is a strong smell of rotting food.

2/10
Here the player finds something that piques their interest.

VGM: On the other side of the room there is a desk. A lot of bookshelves enclose the room. Other than that, the room seems to be pretty clean. You see a small note on the desk.
Player: What does the note say?
VGM: In barely readable handwriting, the note says: "two from right".

3/10
Now the player tries to solve a problem. Note: the player does indeed have a candle with them.

VGM: You go to the cellar. It is really dark down here, you cannot see anything.
Player: I light the candle I have with me.
VGM: The little flame of the candle enlightens the room just enough to have a quick glance. The cellar is very low. The floor, as well as the walls, are brick-built. Everywhere you can see empty boxes and barrels.

4/10
Yes, VGM is also supposed to do battles. So, here is a bit of fighting.

VGM: You are suddenly attacked by a translucent figure.
Player: I attack the ghost with my sword.
VGM: The sword goes right through the ghost.

5/10

Now back to some interesting items that the player finds in one of the rooms.

VGM: You are standing in a narrow corridor without any windows. A well-trodden rug dresses the floor
Player: What does the carpet look like?
VGM: The carpet is a dark red color with a floral pattern.

6/10

Nice, here is another one. Note: the player did indeed find a key in the dining room, that was meant to open the door.

VGM: This room seems to be furnished for comfy evenings. In one corner you can see a brick-built chimney surrounded by an armchair and a small sofa. Abreast the fireplace there is a black piano. You see a door to your right.
Player: I go through the door.
VGM: The key you found in the dining room seems to fit here. You slowly turn it around and the door opens. The room seems to be used as a study. Opposite to the door you can see a desk.

7/10

As the player searches the house, they eventually end up in the cellar.

VGM: The lights on the ceiling make it possible to see everything in the cellar. The room is very low. The floor as well as the wall are brick-built. You can see a lot of empty barrels and boxes
Player: Whats in the boxes?
VGM: There is nothing in the boxes.

8/10

Ok, here is a special one. The player wanted to use their strength. VGM secretly made a skill check to see, whether the player would succeed. They did.

VGM: A pile of ashes lies in the chimney. But underneath there seems

to be a small hatch.
Player: I try to open the hatch.
VGM: With all your strength, you manage to pull the hatch open.

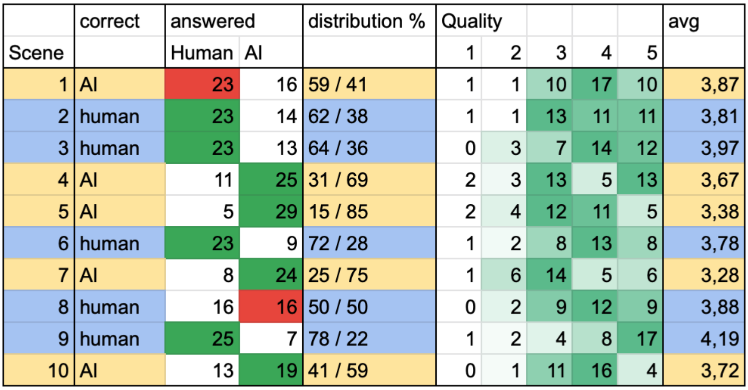
9/10
Another interesting one. As the player walks through the house, they suddenly hear something. They already know there is a piano in the living room.

VGM: Through the silence that surrounded you up until now, you suddenly hear a noise. Fragile and quite at first. Maybe you heard wrong? But after some time you can hear it more clearly. You hear a piano playing a dissonant, monotone melody.
Player: I go to the piano in the living room.
VGM: You move closer to the piano, but the moment you are just a few inches away, it suddenly stops its ghostly play. The resulting silence judges you as if you just interrupted a great concert.

10/10
For the last one, it's getting a bit more spooky. The player is in a house in the woods. They enter a room with curtain draped windows.

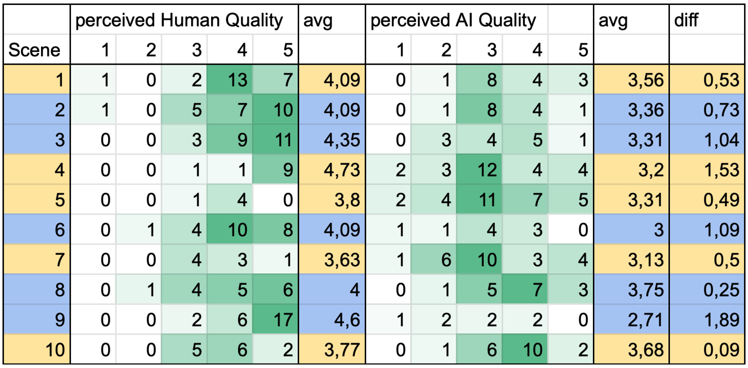
VGM: Behind a dark red, old curtain there is a window through which you can see the mist-shrouded, dark woods.
Player: I pull back the curtain.
VGM: As you pull back the curtain, you see a figure in the distance, watching the house.

12.4.2 Results



	correct	answered		distribution %	Quality					avg
Scene		Human	AI		1	2	3	4	5	
1	AI	23	16	59 / 41	1	1	10	17	10	3,87
2	human	23	14	62 / 38	1	1	13	11	11	3,81
3	human	23	13	64 / 36	0	3	7	14	12	3,97
4	AI	11	25	31 / 69	2	3	13	5	13	3,67
5	AI	5	29	15 / 85	2	4	12	11	5	3,38
6	human	23	9	72 / 28	1	2	8	13	8	3,78
7	AI	8	24	25 / 75	1	6	14	5	6	3,28
8	human	16	16	50 / 50	0	2	9	12	9	3,88
9	human	25	7	78 / 22	1	2	4	8	17	4,19
10	AI	13	19	41 / 59	0	1	11	16	4	3,72

Figure 7: Overall classification and quality rating of all scenes



	perceived Human Quality					avg	perceived AI Quality					avg	diff
Scene	1	2	3	4	5		1	2	3	4	5		
1	1	0	2	13	7	4,09	0	1	8	4	3	3,56	0,53
2	1	0	5	7	10	4,09	0	1	8	4	1	3,36	0,73
3	0	0	3	9	11	4,35	0	3	4	5	1	3,31	1,04
4	0	0	1	1	9	4,73	2	3	12	4	4	3,2	1,53
5	0	0	1	4	0	3,8	2	4	11	7	5	3,31	0,49
6	0	1	4	10	8	4,09	1	1	4	3	0	3	1,09
7	0	0	4	3	1	3,63	1	6	10	3	4	3,13	0,5
8	0	1	4	5	6	4	0	1	5	7	3	3,75	0,25
9	0	0	2	6	17	4,6	1	2	2	2	0	2,71	1,89
10	0	0	5	6	2	3,77	0	1	6	10	2	3,68	0,09

Figure 8: Quality rating split by classification

12.5 Comments

Scene 1

It fits as an working, consistent answer but makes no use of temporal differences nor logical dephts of experiences which could fit in to increase the players experience. It's plenty enough for a use in a small text based AI game but seems to be also limited there.

It's very down-to-earth, not very immersive.

i don't know why but i chose human because of the smelling part.

Informative, yet concise. Just how I like it :)

I think the VGM answer is written by a human because it does include a very specific information about the smell of the kitchen. This implys that the Player should want to investigat that information.

It's a bit brief and not detailed enough. One sink and a table and no more other stuff in there?

It describes perfectly the location, everything you need to know... Tough, I rarely ever heard that word "rickety"

It doesnt flow, it is very rigid

"Small room" and "center of the room" seem disconnnected their respective room scales (like the writer forgot the room was small)

The answer relies on simple key words like sink and table, gets paired with appropriate adjectives under the umbrella term of "old kitchen"

It described the settung / location and also gave hints on what could be investigated next. The last sentence gives a sense of mystery and suggest a way hiw to drive the story foreard.

There is nothing that would spark any doubt about it being written by a human, except the question wether I thought it was written by a human or an AI.

I think the answer isnt detailed enough, so it is for me more likely that an AI wrote it.

Scene 2

Same. And the note itself gotta be useful in some way or else this message in might appear useless as an experience and that would make players lose their attention to details

a bit guessing but I picked AI because i thought I would have said sth like "you pick it up" first. And I would have changed the order of the sentence or use two sentences..

The answer is fine, its just a tiny bit odd to read. Its a bit formal for such a tiny bit of information.

Barely readable doesn't seem to fit a pretty clean room therefore I think the AI is writing the VGM's answer. But because the handnote seems to be holding game progress relevant information the answer could also be written by a human! Either way the answer sounds and feels good.

There'snot much text to begin with, and not much to say. Two from right, sounds a bit strange but I don't have any context here.

sounds interesting

There is no Option for "indistinguishable"

I would assume that a human GM would use more flavor text or some sort of riddle to obscure the grammatically questionable solution

It answers the question directly but also gives some "flavor" by describing the note further by saying its "barely readable".

It straightforwardly answers the question.

I like that it says "barely readable".

Scene 3

Der Satzbau im 2. Satz fühlt sich komisch an aber ich weiß nicht warum. Die Beschreibung hat ne höhere Quali, ist overall länger und beinhaltet auch Formulierungen die eine Erfahrung exakter darstellt und damit vorstellbarer macht

Why is it just “a quick glance” ? Does the candle go out right after? Should it be “in close poroximity?

first sentence. and all in all the sentences stick together to give like a same feeling

Nothing to complain here, sounds fine/legitimed to me

Cellar == room doesn’t seem to fit perfectly. Therefore I think the VGM’s answer is written by an AI that doesn’t know the location it is talking about. The description of the cellar is a bit bland. It does not hold information that stimulates further investigation.

How can I see that every barrel and box is empty if I can only gat a quick glance?

Everywhere you can see empty boxes and barrels. - ¿ Is this even grammatically correct? It sounds off to me.

”that word ””built”” I think a human wouldve choosen a better one”

The last sentence again contradicts the rest. I as a gm would not give fhat Information or say it in concurren e to the beforementioned lack of vision (as far as you can see there is...

I think I missed this with the prior questions but I feel like this answer seems more AI-like because it mostly describes something visual in neutral ways, compared to the question prior to this one that added the bit about ”barely readable handwriting”. To me it sounds more personal to add things that aren’t completely neutral in their description, I think.

The answer provides the answer to a question, that was not directly asked by the player, but is implied in the action the player describes. This seems very impressive if it was an AI

Scene 4

es confused mich dass die translucent figure plötzlich als ghost nicht nur bezeichnet sondern von der reaktion des systems auch als solche erklärt wird. das kann einfach sein dass es einfach ein geist ist aber joa xD über solche sachen denke ich eher nach. Und für nen fight passt die Art von Antwort aber. Ist ja nur ein Beispiel, bisschen flavour oder so kann da ja auch noch dazu kommen XD

very short and direct answer. no description on how the player feels or, sound, surprise, maybe just missing adjectives that are a bit special.

Short, natural, and I get the Information I need. Perfect!

The word ghost from the player is used by the VGM therefore I think it is an AI answering. But because it holds game relevant information (cannot fight a ghost with a sword) it could also be written by a human! Either way the answer itself is smart.

If this is written by AI, it would impressive, considering the AI would somehow know about the intangible nature of ghosts.

boring. the sentence

Like when you repeat the Name of a present after you get it...

The answer is very context heavy, connecting the action with the room it takes place it. I've seen AI do things like that, context sensitivity keeps improving but I just go with my gut and vote human here

It's very coherent with the first and second sentences. I think either there's a way for the AI to pick up on synonyms like translucent figure and ghost or this might have not been computer generated.

A short ans to the point answers. Perhqps this was the AI? I don't know.

I like that is not just damage numbers. I would love if this would be kept up when an attack actually hits, too.

Scene 5

Ähm. ich denke es ist ok für einen Teppich xD

It’s a very simple answer, but the question wasn’t to complex either.

Sounds very cliché, thats why I think its weinten by an AI. A human might have improvised and had a more interesting description.

I think the VGM’s answer is written by an AI because it does not hold game relevant information.

I think the rug

design of that carpet sounds not good

It answers like a person would that has not prepared much of an amswer to that (the hell does it matter?!)

A short description of an item like a rug should be doable for an AI, ideally not from a set list of ”rug addictive” but either from keywords or from image analysis and internet connection

I would not be able to really tell if that was the AI or a human and it passes well in my opinion.

All the information that is needed, I guess.

Scene 6
long answer?

”Meh.

It feels patronizing to not give me a chance to figure out that the Keys I found fits here. I want to figure that out by myself.”

I think the answer is written by a human because it does hold too much specific and game progress relevant information that stimulates further investigation.

The player: I go though the door. As answer, it should be noted first, that this door is locked and you can’t get through, unless you have a key.

trys to built something up

Either one hell of a bad gm or ai. Either way, that answer did a lot of interpretation and additional info dump on the player.

Extremely heavy in context, if that's written by Ai I am deeply impressed

The small descriptions made this answer very natural to me, like mentioning the key being turned slowly or that this room ”seems” to be a study, compared to saying that it ”is” the study.

Perhaps this was a human, because the answer knows the necessity of using the key and that the player owns it already. Though the AI might also know that. So I am not 100

Scene 7
The answer is formed very similarly to the question. I would have liked “they are empty” better, because I know that the answer is about boxes. The word “boxes” doesn't need to be repeated.

short and somehow and i would have expected to be something in there or a bit longer described since it's an important part of the room. also i chose AI because the question is similar to the answer, ...boxes?...boxes

Very normal answer. I am confused by the introduction, because the dialogue didnt mention any noise or piano.

Because the VGM's answer doesn't hold any useful information I think it is written by an AI.

Nothing in all these boxes? That's disappointing.

i think ai wouldve picked randomly something in the box

Question, direct answer, awkward silence. As it should be.

I mean, that's an important thing for the AI to know, when it should reward the players and when not. Which is probably the biggest challenge, because your can definitely work with random drop rates but that can become very frustrating very fast

The answer could be more varied since the boxes being empty has been stated already.

It straightforwardly answers the question. So there is nothing wring with the answer. But it seems like a very basic sentence, making me think this was an AI answer.

Why do they store empty boxes in the cellar?

Scene 8
The words "the hatch" is repeated 3 times in the discourse. Like in my previous explanation.

"picked ai because i thought ai would have chosen the skill check variable and then say if you managed to do it or not.

as well i would have expected sth. about the ashes, dust, coughing, sounds, I don't know. like, there must happen something when the hatch opens when there are ashes on top."

Good answer, and I like that I get to know how much strength I needed to use :)

I think the answer is written by a human because it emphasizes on the strength check that was made.

If you need so much strenght to open a hatch, maybe it could be more descriptive, a bit more atmosphere to it. Sound like hearing the wood crackle, or squeaky hinges, the rust falls off...

The intsrpretation of "all your strength" hints to a close sucess. Dunno if the ai cN do more than yes no in this regard.

Fantastic, the flavour text, the hidden role, very suspenseful

The first part of the sentence describes neatly what happens. So I think this would be a human or if it is an AI, I would be very impressed.

I like that the check is kept secret

Scene 9
It's a very descriptive answer, I like it.

because of the tension the sentence creates, i could image someone telling that. ghostly play - i think ai would have chosen another adjective. last sentence is just great, like just creative :D shocking if it's ai and not human!

Concise, informative and funny. Just how I like 'em :D

I think the answer was written by a human because it is very consistent to the information given beforehand.

hard to read on first go

No buildup on they way there.

Super specific flavour text, sounds very human

It seems slightly inconsistent since the player said they would go to the living room and presumably change rooms, which seems not to fit the answer that says they are simply moving closer to the piano.

Here I am really torn. I think the sentence itself wonderfukly sets the scene making me think of human writing, but the start of the sentence seems as if the player already was in the room with the piano. So this makes me think of an AI, because it seems like the change of location is not properly adressed.

Nice

Scene 10
This is less descriptive, it could have been written a lot creepier, with the player not recognizing the figure at first, for example.

repeating question. and not mutch tension in the sentence, just describing. i mean, there is someone watching the house, thats spooky or at least surprising (i think).

Its fine, but feels a bit lazy.

Because every other question/answer was set within the/a house I think that the VGM's answer was written by an AI and is not game relevant. But it spooky!

not very sure with this one, could be both

Answer looks simple enough for an AI to write this but I still love the answer.

The word to word repetition of what has been done seems to be something the ai never did in the answers before.... Could be just uninspired writing tho... But both seem humane.

I'll go with human, because even though this is perfectly possible for an AI, introducing comprehensive new plot points might be a challenge

It could be either a human or an AI for me therefore I tend to think it was a human, because if I read the sentence and would not know that it was possibly an AI, I would not doubt, that a human wrote this answer.

13 Bibliography

References

Apple Inc. (n.d.). *Speech synthesis — apple developer documentation* [Apple developer]. Retrieved September 16, 2022, from https://developer.apple.com/documentation/avfoundation/speech_synthesis

AXA Group Operations. (2022, September 18). *NLP.js* [original-date: 2018-07-30T17:46:04Z]. AXA. Retrieved September 18, 2022, from https://github.com/axa-group/nlp.js

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., . . . Amodei, D. (2020). Language models are few-shot learners. *arXiv:2005.14165 [cs]*. Retrieved February 18, 2022, from http://arxiv.org/abs/2005.14165

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019, May 24). *BERT: Pre-training of deep bidirectional transformers for language understanding* (arXiv:1810.04805) [version: 2 type: article]. arXiv. Retrieved September 19, 2022, from http://arxiv.org/abs/1810.04805

Edwards, R. (2001, October 14). *GNS and other matters of role-playing theory* [The forge]. Retrieved July 25, 2022, from http://www.indie-rpgs.com/articles/1/

Giacaglia, G. (2019, March 11). *Transformers* [Medium]. Retrieved September 19, 2022, from https://towardsdatascience.com/transformers-141e32e69591

huggingface. (2020). *Summary of the tokenizers*. Retrieved September 19, 2022, from https://huggingface.co/transformers/v4.12.5/tokenizer_summary.html

Kinsley, H., & Kukiela, D. (2020). *Neural networks from scratch in python*.

Koistinen, S., Siukola, M., & Turunen, S. (2005). The process model of role-playing. *In Dissecting LARP*, 205–236.

Mohit, B. (2014). Named entity recognition. In I. Zitouni (Ed.), *Natural language processing of semitic languages* (pp. 221–245). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-45358-8_7

Nakamura, J., & Csikszentmihalyi, M. (2002). The concept of flow. *Handbook of positive psychology.* (pp. 89–105). Oxford University Press.

Nielsen, J. (1994, April 24). *10 usability heuristics for user interface design* [Nielsen norman group]. Retrieved September 16, 2022, from https://www.nngroup.com/articles/ten-usability-heuristics/

Rothman, D. (2021, January 29). *Transformers for natural language processing: Build innovative deep neural network architectures for NLP with python, PyTorch, TensorFlow, BERT, RoBERTa, and more* [Google-Books-ID: Cr0YEAAAQBAJ]. Packt Publishing Ltd.

Schreier, J. (2017, June 20). *The secret behind scribblenauts: Making objects by hand (and lots of crunch)* [Kotaku]. Retrieved September 14, 2022, from https://kotaku.com/the-secret-behind-scribblenauts-making-objects-by-hand-1796262472

Urban, E. (2022, September 15). *Text-to-speech API reference (REST) - speech service - azure cognitive services* [Microsoft docs]. Retrieved September 16, 2022, from https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/rest-text-to-speech

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt, dass

1. ich die vorliegende wissenschaftliche Arbeit selbständig und ohne unerlaubte Hilfe angefertigt habe,
2. ich andere als die angegebenen Quellen und Hilfsmittel nicht benutzt habe,
3. ich die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe,
4. die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfbehörde vorgelegen hat.

Berlin, 30. September 2022