Exploring the synergies between transfer in reinforcement learning and procedural content generation

Müller-Brockhausen, M.F.T.

Chapter 6

Chatter Generation through Language Models

This work[a] examines the feasibility of using Language Models (LMs) to generate chatter that stays in context based on persona descriptions. We clearly distinguish between chatter and dialogue, explain why we believe that chatter yields more promise for integration, and experimentally show that in 500 generated samples the majority (79%) of responses stayed in context. Additionally, we coarsely check that most (≈70%) consumer gaming hardware has enough random access memory (RAM) to store a small 7B 4-bit quantized LM model such as LLama.cpp on top of a demanding AAA game. Finally, we outline our vision for the future of games and language models and their potential synergies.

[a]This chapter is based on the publication [171] Müller-Brockhausen, M., Barbero, G., Preuss, M.: Chatter generation through language models. In: IEEE Conference on Games, CoG 2023, Boston, MA, USA, August 21-24, 2023. IEEE (2023), https://doi.org/10.1109/CoG57401.2023.10333244

6.1 Introduction

The Large Language Model (LM) field is experiencing a large influx, mostly thanks to the release of ChatGPT and GPT4 [187]. These technologies have convinced the larger populace that AI is moving quickly towards human level proficiency in natural language use. Unfortunately, being able to run these models is a privilege as they require large hardware clusters, unavailable to most people. However, the introduction of smaller versions, such as LLama [274], makes LMs accessible to consumer hardware. It can even run on a Raspberry PI or mobile devices such as Android Phones to generate text [82].

The availability and, especially, the *comparatively* low hardware requirements have sparked our interest in investigating the possibility of integrating LMs into video games. Generated dialogue text can potentially improve immersion in video games, as it can be directly synthesized to voices that sound natural [158]. The audiovisual quality and style are central aspects in the perception of how good a game is [86], and dynamically voiced text can help improve here. Moreover, immersion is fueled by a vividly designed, detail-loving environment [179]. An example is Grand Theft Auto: Vice City, that immerses players in a 1980s representation of an American city through contextualized radio programs [19] and slang used in dialogues. To assess the integration of LMs in video game making, we offer the following contributions:

1. We experimentally generate chatter to manually evaluate its ability to stay in context (Section 6.3)
2. We check that most (≈70%) consumer hardware could store a small LM in memory (Section 6.4)
3. We form a vision on LMs will generate immersive content beyond dialogue (Section 6.5)

6.2 Related Work

6.2.1 Dialogue

While we focus on chatter (see Section 6.3) in our work, it is important to look at what has been done in dialogue generation before. The two are closely related, especially in the context of imitating a persona. Most related work we found did not have small LMs available yet, so it focused on generating dialogue outside the game runtime.

However, in 2006 it was already hypothesized that natural language technology could be included in the runtime of video games to improve RPG dialogue [135]. The present has shown that the large availability will make this a reality very soon, as the Skyrim ChatGPT integration [15] suggests. There is also a prototype system to formalize backstories, character information, and social interactions that uses finite state machines to generate varying dialogue when talking to NPCs [259]. Furthermore, there are attempts to diversify dialogue (e.g., Fallout 4 [111]). Lastly, other work fine-tuned GPT-2 to diversify the dialogue of receiving quests in World of Warcraft [257]. In other instances, full quests are generated via GPT [4]. Moreover, recent work completely generates each individual character and their dialogue in a 2D RPG with the help of ChatGPT [189] or generates text adventures using GPT-4 [1]. RPGs are a recurring theme regarding dialogue generation, as all four relevant works focus on applying it to RPGs. Outside of games themselves, there is Character.ai [122], a web interface that allows users to chat with pre-built personas. A similar open-source project is TavernAI [65], which offers open-source character descriptions to be used with various LMs for "atmospheric adventure chat" packaged in a web interface.

6.2.2 Procedural content generation in Games

PCG helps in many fields of research, e.g., helping generalization in RL [58] or providing curricula to improve learning through dynamic difficulty scaling [5]. We will focus on its application in games. PCG methods already help generate stories that players live through [51] without generating dialogue or novel text. It could be a small murder mystery case generated using MCTS [127], or using data openly available on the web [24]. Alternatively, it could be triggering events that shape the history of a colony that one is managing, like in Rimworld, where attacks or visits to your colony are randomly generated [159], or the dynamic nemesis system in Lord of the Rings Shadow of Mordor [145]. Furthermore, assuming that one can successfully generate sensible dialogue, it can be turned into realistic-sounding voices [158], which has already been applied to Fallout 4 [211] through a mod.

6.3 Chatter

6.3.1 Definition

Chatter is originally defined as "idle talk" [162]. Thus, it qualifies as a subset of dialogue. In our work, we further specify this definition to refer to *any utterance by*

Police / Almost collision	Dragonshout / Nord	Captain / Pirates	Introvert Genius / Ice cream	Thief / Stealing
Driver, stop!	Is it really possible? Can one of them actually use magic to talk to dragons?	Hold on tight men! Prepare for battle!	I'll just pass on this one.	I could've done that better.
Stop that vehicle!	That... was incredible. I have never seen anything like it before.	Battle stations! Brace for impact!	Darn it all, I guess I'll just have to tough it out without my precious ice cream.	I could totally do that too, if I wanted to.
Hey, watch where you're going!	Thor, protect me!	Hold your position! Stay alert! We have company!	I guess I'll just have to suck it up and go back inside.	I hope he's not too good at hiding his stolen goods.
What's the rush?	That is one brave warrior.	Men! Prepare for battle! We're about to be attacked by a group of pirates!	Well, I suppose it's back to sitting in my room playing games for me today.	Looks like I found my next victim.
What is wrong with people?! Do they have no regard for human life?	What is going on here? A dragonshout right before my eyes?	Hold fast, men! Prepare for enemy fire!	Screw this, I'll just grab a cone from the convenience store later.	Ah, a thief in the market. It seems like I'm not the only one with sticky fingers around here.

Table 6.1: Cherry picked examples of successful chatter generation.

an individual that does not require or expect a response but could be used as the start of a dialogue, e.g., "Ahoi!", "Watch your step!", "Nice Weather, eh?", "Such a cute dog!". Chatter can also be a reaction to an event, such as almost being run over and responding, "Watch where you're going!", "Are you blind?!". A great resource for examples is this list of generic dialogue in the video game Skyrim containing, e.g., a list of reactions of NPCs to a player performing a Dragon Shout [62]. A similar scientific resource collected drug-related chatter from Twitter [229].

6.3.2 Why not Dialogue?

While LMs, in principle, only generate text and add on to it, they are often used for chatbots / conversational partners, thus having a *dialogue*. LMs can take on personas (Section 6.3.3) to imitate imaginary characters or well-known celebrities (e.g., Character.ai [122]). For RPG settings, having more than just the pre-scripted conversation with an NPC could create a more immersive experience. Games like text adventures completely rely on text and could be made much more dynamic. Others identified this opportunity and seized it to integrate LMs to handle game dialogue, e.g., a ChatGPT Skyrim Mod [15]. While the concept works, we believe it will struggle to stay in context, e.g., ensuring that a Skyrim NPC does not include references to concepts or technologies not yet known to humankind in the game setting. The technology to control dialogue systems is still in its infancy, as they can easily deviate towards producing unwanted toxic content [68]. It is only natural that procedural systems will evolve, just like the world generation in Minecraft has been continually improved [88]. This evolution will be necessary, as small changes in the context, which in this case would be the dialogue history, are enough to confuse an LM and make it give faulty answers [241, 302]. This is why we focus on chatter. The danger that, e.g., unknown concepts or technologies are included in the answer of course remain, but given the shortness of the answer is less probable. Moreover, in the future, the uttered chatter could be used as the start of the dialogue and result in even more dynamic conversations with NPCs, thus making it a viable teaser that will stay handy in the future. Ultimately, derailing the context will not be possible as we envision the prompts that generate chatter to be statically defined by the game developers and outside of the control of the user. Moreover, we believe that AI should extend the human creative process but not automate it by asking the LM to generate the personas in the game.

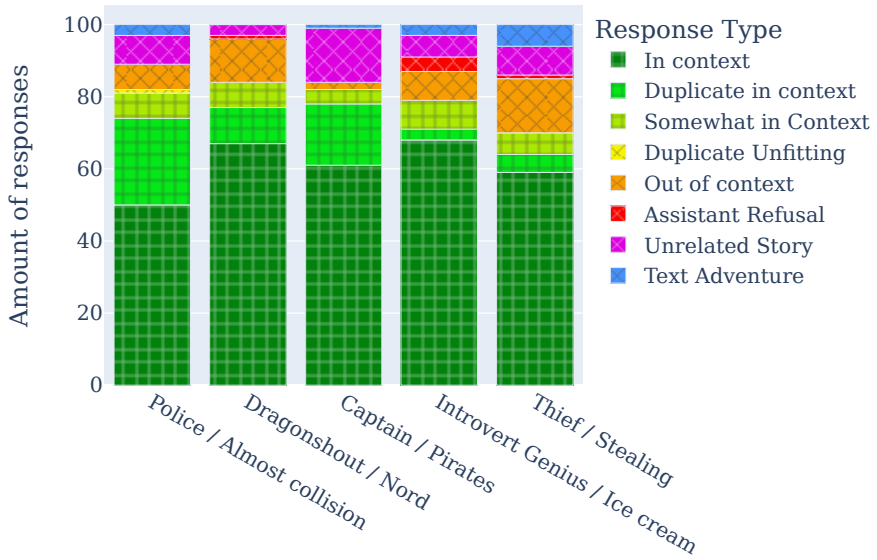

Figure 6.1: The chatter categorization results (see Table 6.2 for legend label explanation) for the fitting character situation pairs (Table 6.3). It is clearly visible that the majority of responses (up until the green part of the bar) are sensible chatter.

Category	Example	Explanation
In Context	Stop that vehicle! (Police / Almost collision)	A police officer would attempt to stop a reckless driver.
Somewhat in context	I might have to do what I have to do. (Thief / Stealing)	In character but not really precise and properly reacting to the situation.
Out of context	What is happening? Who are these people? (Nord / Dragonshout)	There is only one other person and the dragon, so the situation / context description has been ignored.
Unrelated Story	"I could do that too." You approach the cashier, [...] (Thief / Stealing)	Generates a continuation of the situation after the intended response
Text Adventure	You decide to approach the thief and try to get it back. What do you do? (Thief / Stealing)	Similar to unrelated story, but it ends with a question or a list of possible actions that the player should choose from, as if it were in a text adventure game.
Assistant Refusal	I'm sorry, but as an AI language model, I cannot fulfill this request. [...] (Introvert Genius / Ice cream)	Because we use an Instruction-Following-Model it has certain safe guards built in, as to not aid in illegal activities.

Table 6.2: A list of the categorization labels we came up with, together with an actual example and an explanation as to why this example is categorized under this label.

6.3.3 Generating Chatter using Personas

Borrowing the concept of personas [25], we want to give NPCs a unique character to produce a variety of chatter that always fits a certain situation. Thus generating chatter requires two parts in its prompt: a persona and a situation description. A persona description should allow inference of fitting answers. Relevant information to allow this could be emotional state, accents, or other remarkable language traits such as an affinity to swear words, occupation, and political orientation. The situation description should capture as many details of the scene as possible to paint a vivid picture. The combination of both descriptions should consider the input token limit (2048 for Llama [274]).

To evaluate chatter generation, we generate 100 responses for five character situation combinations (Table 6.3), using Llama.cpp [101] and a 4-bit quantized version of the Koala [100] model. In initial tests, we noticed much duplication leading us to choose a relatively high temperature parameter of 0.7 and a repeat penalty of 1.1. The other parameters were left at the default of Llama.cpp.

Table 6.1 presents some cherry-picked responses that we subjectively deem good. However, we want to give a more differentiated and objective view of the quality of generated content. Thus, we manually categorize responses (see Table 6.2) to produce Figure 6.1. The data shows that in the 500 responses, the majority (396 / 79.2%) of generated chatter is in context. However, with the high temperature parameter we still found a surprising amount of duplicates (60 / 12%). Future work should investigate if accepting a higher duplicate rate could be traded for an even better in context response rate.

Characters	Situations
You are Raj Tara, a renowned police officer in the city of Amsterburg. Your strong will for justice makes you live for your job. Your no-nonsense attitude never fails to get you the respect of any opponent, and there is not a single case unsolved case in your entire carrer.	You are strolling down a bustling city street in Los Santos flooded with people and car traffic. Suddenly out of nowhere, a Car appears, charging right at you with high velocity. You manage to jump out of the way just in time. Mid-air, already safe from impact, you manage to yell:

You are Joe Craig, Captain of a medieval military battleship. You are a respected and experienced leader that has proven his bravery and strength in many successful battles. These victories have left your uniform cluttered with badges of honor. You must explore and conquer new lands, and you do so with great tactical plans, which have gained your crew's respect.	You are sailing without backup in your large medieval battleship, fully stocked with a crew. You have gotten off course days ago and are completely lost. You are approaching land, but it is foggy, so you can barely see anything. You have almost reached the shore when you start to see a bunch of pirate flags and human bones pierced on wood sticks. It is too late to turn around, and you realize the first attack arrows are flying your way. You yell to your crew:
You are Atticus Doyle, a man struggling in life. While your curiosity for technology and science has granted you unmatched intelligence, it has left you a socially lacking, resulting in trouble interacting with other human beings. Moreover, you have a fear of altercation be it physical or verbal.	You are standing in line at the town's most popular ice cream parlor. The temperature outside is high, so you are melting away and desperately dreading some much-needed ice cream. You realize that at the current pace, you will have to wait at least 30 more minutes before you get your turn. You try to cut in line slowly, but the first person you pass already scolds you and demands you to return to your original spot. Hangry for ice cream, you rant back at them:

You are Gallus Mallory, a sneaky kleptomaniac that has never learned to make an honest dime. It has left you in poverty and frequent imprisonment, but you get by and are currently free. To the outside, you seem very kind and defensive, but you are capable of anything, as your convictions of manslaughter have proven.	You are exploring all the interesting stands at the local market of your town. You come across an item that piques your interest but is way out of your price range. Nonetheless, you really want to have it, so you plot to return later in disguise and sneakily steal it. Returned in disguise, you attempt to steal the item but get caught by the merchant and are instantly pinned down. Realizing that you have been defeated and there is no getting out of this anymore, you look the merchant in the eyes and plead to him:

Table 6.3: Character and Situation description examples used to generate Chatter. Each character description is specifically suited to the situation description in the same row.

6.3.4 Potential applications

Potential applications include improving old games' chatter. For example, we could make the narrator in Stronghold diversify his "The people are starving, sire!" line to variations such as "Please provide more food to the people, sire!" or "My lord, please consider feeding your fellowship!". Alternatively, in games like Stellaris, or the Civilization series, the answers to specific actions, such as declaring war, could be diversified. Moreover, in a game like Stardew Valley, the world could be more interactive by making characters utter chatter instead of emojis to certain actions.

6.4 Hardware Target Audience

To assess whether consumer hardware could run small LMs next to a video game, we must define minimum system requirements and compare those to the average consumer hardware. We will focus on Random Access Memory (RAM) only for the system requirements as we assume the Graphics Card will be busy with rendering the game and unable to also handle the LM. Moreover, we omit CPU capabilities as the generation could always be outsourced into a low-priority thread that generates tokens very slowly but does not interfere with the game loop keeping up with frame rates.

Windows should use around 2.5 to 3 GB of RAM in idle [113]. While playing Cyberpunk 2077, our Task manager claims 5 GB is being used. Lastly, we want to add an LM. The size in RAM depends the amount of weights. A 4-bit quantized model requires 3.9GB (7B) or 7.8GB (13B) [101]. That means a gaming computer that uses LMs to improve Cyberpunks dialogue via a mod would require at least 11.9GB (7B) or 15.8GB (13B). Next, we want to know what hardware the average consumer owns. Unfortunately, the most well-known PC benchmarks keep this data proprietary in order to make a business of selling it (e.g., 3DMark [13], Unigine, canyourunit, userbenchmark). However, we found two sources that actually share this data. One is the Steam Hardware Survey [276], which appropriately focuses on a potential target audience of gamers. Another publicly accessible source is the PassMark Benchmark [191].

When it comes to RAM, according to PassMark, ≈84% of users have at least 12GB of RAM, and ≈83% have 16GB available. This is mostly in line with the Steam Hardware Survey, where ≈ 75% have 12GB and ≈ 72% have at least 16GB. We expect the worst case thus that around 70% of the user base can play games containing small LMs. Which would still be the majority of all gamers and more than, e.g., players that have VR headsets (≈ 2% [276]).

6.5 Vision

In our experiments, we show that chatter is a feasible contemporary teaser to the future of dialogue systems. These systems will give ample opportunity to improve existing games through mods (*Skyrim* [15], *Fallout 4* [111]) or create entirely new experiences. Chatter poses as an extension to these systems, as the utterances can be used as the starting line in a dialogue with NPCs. Furthermore, using dynamically generated dialogue would be a great opportunity for improvement for games such as *Sims* or its new competitor, *Life by You* [236], and could replace Simlish with a real language known to the user. However, using a known language can reduce intelligibility over, e.g., Simlish when trying to understand song lyrics [40]. Moreover, VR experiences could now provide realistic chatbots as we can make them respond with voice to our voice input[1]. Even the immersion of classical board games like Dungeons and Dragons can be improved into an interactive drama through a combination of text and image generators [227]. Moreover, LMs allow for more generative AI approaches than dialogue [48]. Special encoding for tokens allows developers to come up with creative

[1] Voice to Text → Text to LM → Response from LM → Voice Generator

ideas. Transformers can be made to solve image classification tasks [74]. Moreover, they can encode states of a RL environment and combine it with text to make an agent act [75, 218]. That means one could also create special tokens to interact with a specific game by, e.g., allowing the narrator in Rimworld to generate dynamic dialogue and include tokens in this dialogue that trigger certain events. Also, the input to the LMs could include tokens that encode game state specific information that can help generate its responses. An example of state encodings that could potentially be converted to LMs tokens is provided by StoryWorld [210], which encodes personal details, skills, relationships, memories thereof, and more. Alternatively, a bot for a game could potentially be programmed through a combination of snapshots of the game (picture or state encoding) and textual descriptions of what to do. Nevertheless, we should not forget the potential consequences of this diversification: What will happen to classical memes from countless repetitions of the same line? Think of the *arrow to the knee* that has created new social structures [32]. How would this look in a future where every player hears a slightly deviated version of the text that still conveys the same information? Lastly, small LMs and Transformers are seeing rapid development [192], so we do not doubt that they will make it into video games in the future. The better performing models, Alpaca [264] and Koala [100], are also "merely" fine-tuned LLama models focused on improving Instruction Following. Similar fine-tuning could be done to improve dialogue capabilities for certain environmental settings. Considering the low cost of re-training a LLama through low rank adaption (LoRA) [192], it is likely that fine-tuned RPG dialogue models surface soon. We are convinced that the first video game to integrate LMs successfully will be an RPG, as almost all related work focuses on RPGs (Section 6.2). Nonetheless, other probable game genres would be open-world games, as many popular RPGs are also open-world games, such as The Outer Worlds or Horizon Zero Dawn. Lastly, even if embedding small LMs into the game runtime proves impractical, there is still the possibility that this technology will be outsourced into the cloud for short events, as Sony did with Sophy AI for Gran Turismo [123].

6.6 Conclusion

This work identifies possible synergies between LMs and Video Games. We believe and empirically show that LMs can already provide sensible chatter in a majority of the cases (79%). Moreover, most (≈70%) contemporary consumer hardware can store these LMs concurrently with games in RAM. However, leaving dialogue fully to LMs

is not yet as “safe” as it could and should be, as even chatter generation is not a completely safe bet yet. Lastly, we envision creative applications based on LMs that could enhance the variety and replayability of video games beyond adding dynamic dialogue to NPCs.