Toolkit: Writing for AI

An introductory guide for game writers and makers

Table of Contents

1. Introduction

This toolkit is intended for aspiring games writers and makers. The guide assumes some level of technical understanding and a grasp of the fundamentals of writing in general and for interactive formats in particular, but is aimed at writers with limited or no coding knowledge.

The approaches outlined in this guide are by no means exhaustive or prescriptive and are instead intended as a starting point for those wanting to embark on an exploration of this fascinating emerging form of interactive writing. The main focus here is on conversational narrative games, although some techniques and tools could be equally applicable to other formats.

Writers who do not work or intend to work within games but are nevertheless interested in AI technologies may be interested in the Toolkit *Writing with AI*.

1.1 Using InGAME Toolkits

The remainder of this guide includes 4 main sections, plus a series of additional documents which make up the toolkit. These documents can be mixed and matched along with those from other toolkits to build a custom collection of the resources you need. Words in **bold blue** are defined in the Glossary. If you feel something is missing from the existing toolkits or would like to suggest a topic for InGAME toolkits to cover, please contact: enquiries@innovationforgames.com

1.1.1 In this Document

Section 2

Describes what the guide means by an **AI**-powered narrative and offers some examples to read and play, plus overviews

InGAME Innovation for Games and Media Enterprise

of different approaches, and links to and descriptions of a variety of potential tools.

Section 3
Looks at some of the steps you might take before creating your own AI narrative: finding relevant examples, approaches and tools, and undertaking some exercises to familiarise yourself with these tools to ensure you have made the right choice.

Section 4
Considers a potential writing process for working through an initial draft of an AI conversational narrative from initial concept to early iterations.

Section 5
Draws together the main points from the previous sections and suggests possible next steps.

1.1.2 Additional Documents

AI-Powered Case Studies
(Case Studies 1-3 & 5-6)
Provides case studies of experimental games created during development of the *Writing with* and *Writing for AI Toolkits*, for reference purposes.

Quickstart Guide
Gives a quick overview of the design process used across the Toolkits (and is essentially the **TLDR** version of this document, particularly section 3).

Bibliography
Provides a full reference list and links to all works cited in this and other toolkits, plus additional sites and resources which may be useful.

Glossary
Provides a full glossary of terms used across all toolkits.

Background Information
Gives a brief overview of the research behind this and other toolkits and the background of the researcher who undertook it.

2. What is an AI-powered Narrative and Why Make One?

In the context of this document, 'AI-powered narrative' refers to any interactive story where AI is used to control the flow of information provided to the player. The AI may handle complex character information including personality traits and moods, context-dependent dialogue options, more general variables regarding the current state of the story world, or some combination of these elements.

Using AI in this way allows you to make highly replayable stories without having to manually author huge amounts of text. AI-powered characters can also be more responsive to players and the world state of your game.

2.1 Examples

This section provides a few examples which use different AI technologies in different ways to tell stories. These examples are all free to download at time of writing.

2.1.1 Avery

Avery:
http://mixedbag.it/portfolio/avry/

Format: Mobile

AI System: IBM Watson

Developer(s): IBM & MixedBag

Premise:
In a future where humans are able to fuse with AIs via brain implants, titular AI Avery falls in love with her human 'host', Morgan, a romance which is considered taboo. The player encounters Avery when her memory files have been corrupted and she can recall only her relationship with Morgan and his subsequent disappearance.

Approach:
By chatting to Avery, the player unlocks memory files which help Avery piece together what happened and locate Morgan. While *Avery* technically uses freeform text entry, the player is encouraged to respond directly to leading questions, and via inputting highlighted keywords. This results in a linear narrative. The story is told through a variety of audio and image files, as well as direct conversation with Avery, but text-only options are provided too to aid accessibility. The game contains no fail states as such, although Avery may return that she has not understood an input, or that it is incorrect or irrelevant.

2.1.2 Restless

Restless:
https://inthewalls.itch.io/restless

Format:
Mac/PC (Unity)

AI System:
Spirit AI Character Engine

Developer(s):
Emily Short & Tea-Powered Games

Premise:
The player takes on the role of a ghost haunting the apartment of a young woman called Sylvie. They may befriend or frighten her (and potentially also her girlfriend, Anna).

Approach:
Free-form and non-linear. There is a rough timeline of events that mostly happen every playthrough, although how they occur can vary depending on the player's interactions with Sylvie. The menu of player dialogue options is generated on the fly and can be altered by changing the player-character's mood. Multiple moods can be selected at once, generating different menu options. Keywords can also be discovered and pursued as topics of conversation. There are multiple endings and pathways through the narrative. There are no fail states.

2.1.3. The 3% Challenge

The 3% Challenge:
https://doppio.games/three-percent

Format:
Mobile/Amazon Alexa/Google Assistant

AI System:
Amazon Alexa/ Google Assistant

Developer(s):
Doppio Games

Premise:
A prequel to the Netflix television series *3%*. A sci-fi thriller in which twenty-year olds must undertake 'The Process' an experiment which determines who is worthy of leaving their impoverished homes behind and beginning a new life in a wealthy utopia.

Approach:
The AI voice assistant handles spoken or typed player input and NPCs respond accordingly. Players often unwittingly make decisions about who to form alliances with simply through speaking to one character rather than another, or siding with one over another during an argument. There are various puzzles and challenges relating to listening to sounds or memorising sequences, and these can be failed resulting in game over.

2.2 Approaches

As you can see from these examples, there's no single approach to writing an AI-powered narrative. The AI may handle the behaviours of a single character (as in *Avery*), or multiple characters (as in *Restless*), or it may act as the overall framing system which controls the delivery of all elements of story and gameplay (as in *The 3% Challenge*).

The story may be fairly linear, with the overall narrative remaining broadly the same regardless of player inputs, or the way the player interacts with the characters may have an impact on how things turn out, with multiple possible endings and pathways. Players may choose pre-defined options to respond or the game may allow free-form text entry.

The game may record a large amount of information about the player's responses, or it may only react to the most recently entered phrase. In order to create more tailored outputs, the game may have specific pre-authored responses relating to specific keywords, or it may generate responses according to what it knows about the player and game-world. The former is more labour intensive, but is more likely to give coherent responses, the latter makes it easier to generate larger volumes of content, but is more prone to incoherence and unexpected behaviours. Most games attempt to blend both techniques to some degree for the best results.

2.3 Potential Tools

This section provides a non-exhaustive list of potential tools for creating AI-powered narratives, and describes some of their features in order to help you decide which might be most suited to your needs.

2.3.1 Charisma AI

Charisma AI is a tool for creating AI-powered interactive comics which are published via its mobile app. It allows writers with no coding experience to create stories with artwork and audio using a simple visual interface.
It's well-suited to narratives with multiple characters as any number of non-player characters can be created to interact with each other and the player.

Charisma AI:
https://charisma.ai/

Output Format: Mobile app

Author Interface:

- Visual – narrative is created through linked Player and Character nodes
- 6 player intent pre-sets (thumbs up, thumbs down, don't care, confused, compliment, insult) to help recognise & respond appropriately to player input
- 5 pairs of opposing character moods (sad/happy; calm/angry; untrusting/trusting; impatient/patient; brave/scared) which can be adjusted on sliding scales to adjust responses

- Categories to handle synonyms for key words or groups of potential related inputs
- Memories and gates to record variables (string, Boolean and numeric) and control narrative branches
- Character creation tools allows creation of cartoon avatars for in game characters
- Media tool allows usage of pre-existing backgrounds and audio, plus upload of new items

Player Interface:

- Natural language text entry, with text to speech capability

Support:

- Discord server
- Detailed sample and quick start template
- Full documentation
- Tutorial videos

2.3.2 Actions on Google

Google Actions is a development tool for creating conversational games and narratives for Google Assistant. It requires at least intermediate understanding of JSON in order to properly understand its processes, although a writer working with development team support would probably manage with only rudimentary JSON knowledge. While this may prove challenging for individuals or freelancers, the enormous install base of potential users with Google Assistant on their phones could make for a good opportunity for the right team.

Actions on Google:
https://developers.google.com/assistant/conversational/quickstart

Output format: Google Assistant (mobile phone/PC/mac)

Author Interface:

- Split across 3 platforms, the Action console & the Dialogflow console (both online menu-based systems) and Firebase (a downloadable command line tool which handles the code for the project)
- Recently updated (2020) to include a visualisation tool which aids with conversation flow design and simplifies and consolidates some of the JSON elements allowing for code edits within the one interface
- Includes a testing environment and debugging tools

Player Interface:

- Natural language (voice-controlled or text entry), menu-based or a mix of these
- Chip functionality (text prompts)
- Carousel functionality (image prompts)

Support:

- Templates are available for common features
- Detailed online tutorials on a wide range of topics

2.3.3 FAtiMA Toolkit

FAtiMA is an open-source tool for creating AI-powered characters. It requires a deep knowledge of AI systems, conversational structures and variable management, although it does come with robust documentation and video tutorials.

FAtiMA Toolkit:
https://fatima-toolkit.eu/1-what-is-fatima-toolkit/

Output Format: Testable within the Integrated Authoring Tool. Comes with a template for Unity, but due to being open source, is theoretically compatible with any game engine.

Author Interface:

- Integrated Authoring Tool creates a Scenario file which stores character, world and dialogue information and a Cognitive Rules file which records all the rules and variables which determine character actions
- Characters can be created with an initial Knowledge Base (their thoughts, beliefs and known information), Emotional State (characters may experience varying degrees of paired emotions: resentment/happy-for, pity/gloating, hope/fear, pride/shame, admiration/reproach, joy/distress, gratitude/anger, relief/disappointment and satisfaction/fear-confirmed. They can also express feelings towards objects ranging from love to hate.
- The Emotional Decision Making system allows for the assessment of stats in the various above categories in order to determine dialogue choices
- Social Importance allows allocation of how important something is to the character in order to rank decisions
- Comme il Faut is a system which determines which social actions characters should take according to social expectations (for example, greetings, farewells, responses to insults etc)
- Dialogue Manager allows writers to author the text characters will speak and set the conditions under which they will speak using a series of tags and variables. It also contains a map view in order to allow ease of checking narrative pathways
- World Model allows the writer to set not only dialogue actions, but also other actions the character may make (movement, animation etc) although these will need to be coded in the game engine too in order to work.
- Metabeliefs are a further tool for determining what actions characters should take according to their beliefs and previous actions.

Player Interface:

- In-game testing via the Simulator allows clicking of links to choose action and/or dialogue options for any character
- Unity template includes a menu-based system which displays available options

Support

- Website includes videos, playable demos and written documentation
- An in-tool tutorial is provided (although it is easier to

understand after completing video tutorials)
- The project Github page includes further information and assets

2.3.4 Spirit AI Character Engine

Spirit AI's Character Engine (CE) was a tool aimed at writers with a good understanding of conversational narrative and logic. It could be integrated into games made with Unreal and Unity to handle dialogue options and character personality. It had no standalone output formats, but came with templates that would allow solo developers to create simple Unity scenes. As of 2021, it has ceased development and is no longer available to download. It is included in this document for reference only. For those interested in a tool of this kind, CE's base authoring tool, Expressionist is still available.

Spirit AI Character Engine:
https://web.archive.org/web/20210115183354/https:/www.spiritai.com/character-engine/

Output format: Unity/Unreal for Mac, PC or console. No mobile support.

Author Interface:
- Script view for main dialogue options, menus and databases (with a combination of drop-down choices and free form text entry) to handle all other information
- 28 player intent presets covering a variety of questions (e.g. where, when, what, who) and statements (e.g. insult, compliment, greeting, confused)
- 5 preset traits which are **Booleans** (angry, anxious, calm, happy, sad) although any number traits can be created and may be set as Boolean, **string** or numeric
- Entities which can be fully edited to add large amounts of information about anything relating to the game world, characters, events etc
- Tags which can be used to **gate** information or prompt conditional dialogue
- In-tool testing environment

Player Interface:
- Natural language text entry or menu-based
- Unity and Unreal integration allows customisation as required by the development team for virtually any kind of interface

Support:
- N/A

3. Experiments and Exercises

Obviously there's a huge amount of variation between types of AI narrative in terms of tools, authoring interfaces and player interfaces which means there's no single workflow that will work for every project. However, there are steps you can take to help refine your narrative ideas and define your workflow according to your project needs. The following exercises are to help you establish how your AI narrative will take shape. You may want to work through them in order, or jump ahead to the one that most suits the stage you are at in your project. If you've already chosen a tool, you may wish to skip to section 4.

3.1 Reviewing Examples & Approaches

A good way to get started with any project is to look at some existing examples and consider how similar or different you'd like your own project to be. Below, we work through the methodology provided in the Quickstart Guide step by step.

3.1.1. Exercise 1 – Reviewing with Purpose

Find some AI-powered narratives either using an internet search, the 'inspiration' section of the Bibliography or the examples provided in section 2.1. Once you've found at least two that interest you, play through to get an initial impression, then consider the questions below:

What was the AI-powered character like?
How was their personality conveyed?
What was the story?
Were there any parts you missed or didn't understand?
If so, why did this happen? How could it be improved?
If not, how do you think this was achieved? What might have made it even better?

In answering these brief questions, you should now have some idea of features that work well, or perhaps elements that consistently fail! Play and review more examples until you're happy you have a sufficient understanding of what makes a good AI-powered narrative.

You may also want to compare some of the games to see whether some solved the same problems in different ways, or handled different elements better or worse than one another.

3.2 Selecting Tools

There are two potential routes to selecting an appropriate tool. One is to review the needs of your project and select accordingly. Another is to experiment with several tools (see section 3.3), choose the one you are most comfortable with, and design your project with that in mind.

> *"Constraints help develop creativity in beginners and help experts structure and solve creativity problems."*
>
> *Patricia D. Stokes (2006) Creativity From Constraints: The Psychology of Breakthrough. New York City: Springer Publishing Company. p 131.*

3.2.1 Exercise 2 – Designing by Constraint

Setting **constraints** can be a useful way to move forward with design, limit scope, choose an appropriate tool, and come up with creative ideas and solutions, rather than merely repeating what you've seen before.

If you've completed Exercise 1, or have already started thinking about how you might want to use AI within your game, you may have some constraints in mind.

Example constraints:
The game must use menus rather than natural language.
The AI character must begin in an angry mood and move towards a happy mood.
The player must remember and repeat phrases uttered by the AI character.
The game must be playable on a mobile phone.

Use your own, or select a couple from the list above. Now consider how these constraints will affect your game, both in terms of its narrative and gameplay. Ask yourself:

What tool would work best given these constraints?
What device would make best use of these constraints?
What do these constraints suggest about the character and their story?
What do they suggest about the role of the player?

Perhaps you've already chosen a tool, and that suggests constraints of its own. Consider what they are and put them to use.

3.3 Experimentation

While experimentation shares some similarities with prototyping, it is a distinct design phase in its own right. A prototype would generally be a precursor to a more fully realised product. An experiment is intended purely as a learning experience. While the output of an experiment may later be developed further, there's no expectation that it will be.

3.3.1 Exercise 3 - Design your experiment

In order to ensure your experiment provides sufficient information, you should spend a little time planning it. Ask yourself the following questions:

What are the features of this tool which I want to explore? If you have already undertaken exercise 2, you may not need to test all the features. Even if you are simply seeking to get a general sense of the tool, it is probably best to select a few key features to prevent the experiment from becoming overlong or complex.

How long am I prepared/able to spend?
Your experiment should be as short as is possible. Remember it is for learning purposes, and therefore does not require polish.

What story am I going to tell?
Rather than spending a great deal of time labouring over details, it is more efficient to use the concept you designed in Exercise 2 (which will also allow you to evaluate it), use a well-known story such as a fairy tale, or adapt an existing story or game. This will prevent you from getting bogged down in narrative detail, but will still allow you to gauge how well the tool allows you to communicate a story.

3.3.2 Exercise 4 – Experiment

Do the experiment!
Do not extend the timespan you have allotted. - If you uncover additional features within the tool, make a note of them and move on.

3.3.3 Exercise 5 – Evaluate

When evaluating what you've made, it's important to remember that you're not evaluating how 'good' it is. It may not be finished. Elements of it will probably not work! The aim of the initial experiment phase is to familiarise yourself with the tool, and so your points of evaluation should be focussed around that:

How easy was the tool to use?
Did the key features work as expected?
If some elements took more time to grasp, were they worth the effort?
How useable was the output?
e.g. Did you have to go through multiple steps in order to have a playable, functioning item? Did you have to use other tools and systems?
Was it possible to view and test what you had made as you were making it?

Is the output in line with your project's requirements?
Were you able to tell your story?
You can be as brief or detailed as you like in your responses, but together they (along with the experience itself) should give you an idea of whether this is a tool that warrants further exploration or if it's not suited to your needs.
If you are undecided, return to Exercise 3 and design a new experiment to explore additional features of the tool.

If this is the tool you want to progress with, take some time to note down your workflow. What order makes the most sense? How early into the flow should testing be introduced? At what point(s) do you need to use other systems and/or tools? This will give you an initial workflow and help you speed up your processes when returning to the tool.

4 Writing Process

Up until now, this toolkit has considered some fairly general experiments and processes suitable for any tool. Now it will focus in on the specifics of writing for AI-powered stories and games. Not every method will be suitable for every type of work and this is not intended to be an exhaustive list – it should instead provide some starting points to aid you in developing your own writing strategies and processes. You may want to use a single method or several within one project. Note that most of the methods here focus on character and conversation. This is because most AI-powered narratives use the AI-character and their dialogue as the main driver of the story. However, if you'd like to try something a little different, the Text Generation Toolkit and related Case Studies might have some useful examples.

4.1 Drafting

It's probably advisable for at least some of your drafting to occur outside of your chosen tool. This will allow you to make mistakes and work out your characters and story without investing a lot of time in coding. However, if you're very experienced, spent some time designing experimental pieces or are making a short and simple game, you may wish to fast forward ahead to Prototyping in Section 4.2.

4.1.1 Method 1: Persona Creation

As the name suggests, Persona Creation involves developing a **Persona** for the character(s) the user will interact with. This includes detailed consideration of their personality, including likes and dislikes and sample snippets of dialogue which capture the character's tone, attitude and any verbal tics or habits. This design method is particularly useful for games and stories where the main focus is the relationship between the player and the AI character.

Google has in depth design guides for creating Personas along with templates. However, Personas are more often used in connection with business (e.g. to develop the personality of 'helper' agents and online assistants), and therefore additional consideration may be needed in order to transpose them to a narrative context. There are other useful resources relating to this in the Bibliography.

4.1.2 Method 2: Conversation Flow

Conversation Flow is a catchall term for how a typical conversation with an AI character might play out. Again, Google have a variety of resources for designing conversation flows, many of which involve

creating simple flow diagrams. These diagrams are useful for getting a sense of how your story might run from beginning to end. The main drawback here is that these flows often only include best- or worst-case scenarios (e.g. the player fully co-operating, or attempting to totally de-rail the narrative) and therefore you may find some blind spots or gaps when moving to the implementation phase, particularly if you are using a lot of branching choices. However, this needn't be a problem as long as you are aware when implementing. Examples are also usually geared towards commerce rather than narrative, so additional examples from non-AI videogames may be of use here. (See both sets of Case Studies for more).

4.1.3. Method 3: Draft Database

Many narrative designers in the games industry spend far more time writing in Excel spreadsheets than expected. Depending on the nature of your story, an Excel spreadsheet can be very useful for drafting your AI-powered narrative as you can use it to start building a database of the **knowledge model** your AI character(s) will need to have. An example Excel database is given in the AI-powered Case Studies document and the Github Repository. This method may be more useful in stories where the main focus of the narrative is uncovering a mystery or finding out more about the story's world, inhabitants and events.

4.2 Prototyping

Prototyping may be considered another form of drafting. The key difference between the drafts in the previous section and the prototypes described below is that the latter result in a playable piece. You may wish to expand one of your earlier experimental pieces to create your prototype.

4.2.1 Twine Prototyping

If you began creating a Conversation Flow diagram for your AI-powered game, you may have found yourself wishing you could add more complexity to the branching, or perhaps include some conditional variables.

Prototyping in Twine serves a dual purpose. It allows you to expand your flow diagram quickly and provides a clear visual representation of it which is easy to edit. Twine also allows you to create a playable approximation of the final experience without committing large amounts of development time. Obviously, it does not have the level of complexity and power available in an AI-powered system, but it will allow you to present beta testers with an approximation of the experience you hope to achieve. You'll also be able to gauge their reaction to both the writing and the overall narratives and make adjustments before implementing into the system you're actually using.

4.2.2 In-Tool Prototyping

Sooner or later you'll need to start adding your work into the actual tool you've chosen for your project. This is where having done some initial experiments or other drafting will come in very useful.

The main thing to remember is that at this stage, you are simply trying to get the main structures of your story and characters in place. Test often, and get beta testers involved as soon as you have a version which is playable from start to finish. It's tempting to try to polish everything before letting others play, but

as with all interactive works, watching someone who isn't as familiar with it as you will provide invaluable feedback, even if their comments aren't particularly helpful. They may suggest changes and improvements you are already intending, but they're also likely to highlight conversation pathways you haven't thought of or find unexpected issues, even if it is through their actions, rather than their feedback.

4.3 Implementing and Iterating

Once you have a fully playable prototype from beginning to end, the process is really no different to any other game develop cycle – continue to play and improve your piece. Accidentally saving over your current draft could cause problems or loss of work (and with autosave functions in some of the tools, it's easily done). Therefore if you aren't already used to working with a versioning tool such as Perforce or Github, you should consider making use of one. Github is a good starting place for complete beginners.

4.4 Key Takeaways

In the AI-Powered Case Studies, you'll find examples of all of these various processes. However, for convenience, here are a few of the main points to remember:

- Be aware of the technological limitations of the tools and systems you are using and build those limitations into your narrative design
- When choosing keywords or tags, consider not only what they are, but how they might be used – do they have dual meanings? If the player uses a synonym, could it accidentally trigger a different part of the narrative?
- Bear in mind that audiences are often eager to interact with machines and will assume additional intent and personality if you give them enough to go on

5 Conclusions

This toolkit began with considering what AI technologies are, what they can do and whether they are relevant to your project. In order to help you decide, it provided some examples of AI-driven games, and a framework for analysing them in order to help drive your own project forwards. It provided some suggestions for AI-powered tools, systems you could use, and some experiments for getting to grips with those tools (and any others you may find). Finally, it suggested some processes and techniques for the actual business of writing your AI-powered narrative.

The accompanying documents (particularly the Bibliography) contain further reading which may be useful. If you're particularly interested in writing, the toolkits *Writing with AI* and *Writing with Text Generators* may also be of interest.

Good luck with your AI-powered project!