Mastering AI Dungeon Scenario Creation: A Deep Dive for Gemini Gem Development

Executive Summary

AI Dungeon represents a sophisticated intersection of interactive storytelling and generative artificial intelligence, providing an "AI-guided role play experience". Its core functionality relies on Large Language Models (LLMs) to produce "infinitely generated text adventures". The efficacy and coherence of these narratives are critically dependent on the design and strategic application of various scenario components. This report provides a comprehensive analysis of these elements—including the opening Prompt, AI Instructions, Plot Essentials, Author's Note, Story Cards, and Triggers—and elucidates their interplay with the underlying AI models. The objective is to furnish detailed information that will be instrumental in training a new Gemini Gem, enabling it to effectively assist in the creation of robust and engaging AI Dungeon scenarios. The analysis underscores the necessity for the Gemini Gem to comprehend the intricate dependencies between components, the inherent limitations of AI, and the strategic techniques required for optimizing AI outputs.

1. Introduction to AI Dungeon Scenarios

1.1 What is an AI Dungeon Scenario?

An AI Dungeon scenario functions as a foundational template, or "predefined Scenario," that players utilize to initiate a new adventure. It is the starting point from which a unique, AI-driven narrative unfolds. When a player selects a scenario, its pre-configured elements, such as Plot Essentials, Story Cards, and other settings, are transferred to the new adventure, establishing an initial narrative framework. This process transforms a static template into a dynamic "seed environment" that primes the underlying LLM for a specific narrative trajectory, tone, and world. The quality of this initial seeding directly influences the coherence and desired direction of the subsequent "infinite" generation. The AI, acting as a "predictive text bot" , leverages this initial context to continue the story, making the scenario's design paramount to the adventure's success.
Scenario creation is managed through the "My Stuff" page, where users can access a dedicated creation page filled with various fields for customization. These fields allow creators to define the adventure's initial conditions, guiding the AI's generative process.

1.2 The Role of Scenarios in AI-Guided Interactive Storytelling

Scenarios are more than mere static templates; they are dynamic "seed environments" that prime the underlying LLM for a specific narrative trajectory, tone, and world. The quality of this initial seed directly impacts the coherence and desired direction of the "infinite" generation. The AI doesn't simply read the prompt and cease; it uses it as a foundation to generate the remainder of the story. Consequently, a scenario is less akin to a fixed script and more like a carefully constructed "seed environment" that influences the LLM's ability to extrapolate a

consistent and engaging narrative. This implies that for Gemini Gem training, understanding scenarios as dynamic seeds necessitates that the Gem learns not only what content to include but also how to structure it to maximize the AI's capacity for coherent and engaging narrative generation. It is about establishing the initial conditions for a complex, evolving system.

1.3 Overview of Scenario Creation Workflow

The scenario creation process in AI Dungeon involves populating several distinct fields, each serving a unique purpose in guiding both human players and the AI.
The **Title** of a scenario serves as its primary identifier within the "My Stuff" section and in public search results. It must be unique and descriptive to facilitate easy discovery by creators and other players.
The **Description** provides more detailed information about the scenario. The initial lines are visible in search results alongside the title and tags, making them crucial for attracting players. The description is primarily intended for human users; the AI does not process this text. This distinction is significant: while the early parts of the description are for discoverability, later sections can be utilized for providing instructions, explanations, or notes directly to the player, without consuming valuable AI context. This highlights a critical design consideration: some components are primarily for human users (e.g., discoverability, player instructions), while others are direct inputs for the AI. The Gemini Gem must be trained to differentiate between these audiences, optimizing for human readability and guidance when generating descriptions, and for precise, AI-interpretable language and structure when generating AI-facing components. This necessitates distinct modes for human-facing versus AI-facing text generation within the Gem.
Scenario Options represent an advanced feature, enabling creators to build selectable choices at the beginning of an adventure, similar to the Quick Start menu. Each option can lead to a sub-scenario, complete with its own Prompt, Plot Essentials, and other settings, allowing for branched narratives or customizable starting points.
Tags are keywords added to scenarios to improve their searchability and categorization, helping players find content relevant to their interests.
Flagged NSFW and **Publish/Unlisted** settings control the visibility and compliance of content with community guidelines. Flagging NSFW content ensures it is hidden from safe searches, maintaining platform integrity. Publishing makes a scenario publicly discoverable, while "Unlisted" allows sharing via direct link without appearing in search results. Creators are advised to thoroughly proofread and test scenarios before publishing.
The following table summarizes the key scenario components, their primary purposes, AI visibility, and strategic applications:

Component Name	Primary Purpose	AI Visibility	Key Characteristics	Strategic Use Case
Title	Human Search/Listing	No	Unique, descriptive name	Discoverability, setting player expectations
Description	Human Search/Listing/Info	No	Detailed info for players, first lines for search results	Player instructions, creator notes, marketing
Prompt	Initial narrative injection, first action	Yes	"Meat" of scenario, starting text; can be max size but top fades quickly	Setting immediate scene, character's starting state, inciting incident
AI Instructions	Global guidance	Yes	Powerful, general	Overarching

Component Name	Primary Purpose	AI Visibility	Key Characteristics	Strategic Use Case
	for AI behavior, style, tone		directives; positive phrasing preferred	narrative control, avoiding repetition/clichés
Plot Essentials	Persistent core memory for always-relevant details	Yes	Always loaded into context; concise, action-relevant	Character traits, setting, overarching plot points, crucial lore
Author's Note	Immediate/local guidance for genre, style, tone	Yes	Inserted near bottom of context, bracketed; direct influence on next output; short (3-4 sentences)	Fine-tuning narrative style, specific scene instructions, theme reminders
Story Cards	Conditional memory for specific, triggered details	Yes (when triggered)	Loaded when keywords appear, prefaced by "World Lore:"; lower priority, can leave context	Detailed lore, character backstories, location specifics, item descriptions
Triggers	Keywords that activate Story Cards	Yes (implicitly)	Case-insensitive but sensitive to spaces; comma-separated	Linking specific narrative elements to their detailed descriptions

This table provides a high-level reference for all scenario components. For a Gemini Gem, this summarized understanding is essential for correctly categorizing user input and generating appropriate content for each component. It clarifies which elements are processed by the AI versus those intended for human consumption, and the primary function of each, which is foundational for effective scenario creation assistance.

2. Core Scenario Components: Design and Best Practices

2.1 Opening Scenario/Prompt

The **Prompt** serves as the initial narrative injection, representing the "meat" of any AI Dungeon adventure. It is the first message the player encounters and the foundational text the AI utilizes to generate its initial response. The prompt can be filled to its maximum size, providing the AI with a substantial initial context.

Effective prompts are characterized by their specificity, descriptive language, and the use of action verbs. For instance, instead of a vague instruction like "I want to fight a dragon," a more effective prompt would be "I want to fight a red dragon with black scales and a long tail". This level of detail enables the AI to better understand the user's intent and generate more vivid and relevant responses.

A critical best practice involves avoiding the use of negative phrasing, such as "not," in prompts. The AI frequently misinterprets such negatives, often leading to outputs that contradict the creator's intention. Similarly, while descriptive language is beneficial, excessive specificity, particularly with details like colors, can be counterproductive. The AI often struggles with

maintaining consistency in these highly specific areas, necessitating frequent manual edits by the player during gameplay.
For a more immersive experience, employing the second-person point of view ("You...") is recommended. Furthermore, ending the prompt with a leading verb (e.g., "You awaken," "You find") can immediately stimulate the AI to generate an engaging continuation of the story. Prompts can also incorporate user-entered fields, denoted by a dollar sign followed by curly brackets (e.g., ${What is your gender?}). This feature allows players to customize their adventure at the outset, enhancing personalization.
A key observation regarding the Prompt is its role as an "ephemeral foundation." The prompt is the "first message" and "first Action," but any information placed at its very beginning "will disappear from the Current Context relatively quickly in a long Prompt". This indicates that while the Prompt is crucial for *initiating* the narrative, its direct influence on the AI's *ongoing* generation diminishes rapidly as the story progresses and new text pushes older content out of the limited context window. Plot Essentials, by contrast, are explicitly recommended for "tone-setting" information that needs to "always stay at the top of the Context". The limited context window is the underlying cause for this phenomenon; the Prompt, despite its initial importance, becomes "forgotten" by the AI over time. Therefore, the Gemini Gem should be trained to understand the Prompt's role as a strong *initial* catalyst rather than a persistent memory. It should prioritize setting the immediate scene, the character's starting state, and the inciting incident. For long-term narrative elements, it should strategically advise placing them in Plot Essentials. This requires the Gem to comprehend the "decay" of information within the LLM's context.
Another observation highlights the Prompt as a "micro-prompt engineering" exercise. The specific linguistic advice provided—"be specific," "descriptive language," "action verbs," "avoid 'Not'," "use second person," "leading verb" —are fundamental prompt engineering techniques applied at the level of the initial input. The AI's tendency to generate better responses with action verbs or its "quirks" like misunderstanding "not" are direct manifestations of its training data and probabilistic language processing. This underscores that even within a specialized platform like AI Dungeon, core LLM prompting principles are applicable. The Gemini Gem, when assisting with prompt creation, must internalize these granular linguistic rules to generate effective initial inputs. This suggests the Gem needs a deep understanding of how specific phrasing impacts LLM behavior, enabling it to offer precise, actionable advice on prompt construction.

2.2 AI Instructions

AI Instructions are a powerful feature providing unprecedented control over the AI's behavior and response generation. They function as a set of global directives that steer the AI in the desired narrative direction, influencing elements such as writing style, conflict level, story events, character behavior, and point of view.
These instructions can be added during scenario creation or to an existing adventure via the "Add Plot Component" button. While optional, well-crafted AI Instructions significantly contribute to maintaining narrative coherence and consistency. They are particularly effective in addressing common AI issues, such as avoiding cliché or repetitive phrases and other common tropes. For example, instructions can specify a "H.P. Lovecraft vibe" or dictate that "only use technology, tools, settings, and locations that would have been present during the Middle Ages".
A crucial aspect of AI Instructions is the importance of positive phrasing. AI language models operate like "probability machines" and tend to struggle with negative instructions. Telling the AI "don't talk about this" can inadvertently make it focus on the forbidden topic (e.g., "don't think about the blue banana" might make it harder to ignore blue bananas). It is more effective to instruct the AI on what it *should* adhere to, rather than what it should avoid.

AI Instructions are categorized into Model Default, Scenario Default, and Custom types. Model Default instructions are built-in for specific AI models. Scenario Default instructions are set at the scenario level and apply to all adventures created from it. Custom instructions are user-defined or modified instructions. Any edits to default instructions automatically convert them to custom. A blank custom instruction component will behave differently than having no instruction component at all, which defaults to model instructions, potentially leading to unexpected story outcomes.
The placement of AI Instructions within the AI's context is strategic. They are included towards the beginning of the context , giving them a foundational influence on the AI's generation. This positioning allows them to set overarching rules and stylistic guidelines that the AI considers throughout the narrative. This is distinct from Author's Note, which is inserted near the bottom of the context and has a more immediate, direct influence on the current output. The comprehensive nature of AI Instructions, coupled with their early placement in the context, makes them highly potent for general direction. This implies that the Gemini Gem should prioritize the generation of clear, positively phrased AI Instructions for establishing the fundamental narrative parameters, reserving Author's Note for more immediate, scene-specific adjustments.

2.3 Plot Essentials

Plot Essentials (formerly known as "Memory") are a fundamental feature designed to provide the AI with critical story details that it should consistently consider during the generation of every part of the adventure. This component addresses the inherent limitation of LLMs, where older narrative elements can be "forgotten" as new text fills the finite context window. Plot Essentials ensure that crucial information remains at the forefront of the AI's awareness, promoting coherence and consistency.
This component is always loaded into the AI's context, regardless of recent mentions in the story. This makes it ideal for information that is perpetually relevant to the narrative, such as core character traits, the overarching setting, or persistent plot points. Examples of effective content for Plot Essentials include:

- Information about the player's character: personality, goals, companions, and what they are known for.
- Details about the local area or setting: landscape, politics, weather, and unique cultural quirks.
- Physical descriptions that are consistently relevant, such as a character's strength or a specific item they always carry.

When crafting Plot Essentials, conciseness and density are paramount. The aim is to pack as many relevant details as possible into minimal space. Only information intended for immediate and continuous reference by the AI should be included. Information in Plot Essentials "will prime the AI to bring up those details again" , so creators should be selective, ensuring that only desired elements are consistently highlighted.
Similar to AI Instructions, Plot Essentials should avoid negative phrasing. Instead of stating what is "not" the case, it is more effective to use affirmative language or words like "avoid". To enhance word association and ensure the AI correctly links information, individual topics should ideally be on their own line, without extraneous details, and the subject's name should be repeated a few times.
Plot Essentials are considered "Required Elements" in the AI's context assembly. They are given high priority, included after Front Memory and Last Action, and before Story Summary, if space permits. In the final context sent to the AI, Plot Essentials are positioned directly after AI Instructions. This ordering reinforces their fundamental role in establishing and maintaining the core narrative framework.

The constant presence of Plot Essentials in the context makes them distinct from Story Cards, which are only loaded conditionally when triggered by specific keywords. This difference is crucial for effective context management. For instance, if a character is a constant companion, their personality and key traits should reside in Plot Essentials to ensure the AI always references them. Conversely, details about a character who only appears occasionally, or specific lore that is only relevant when mentioned, are better suited for Story Cards. This distinction ensures optimal token usage within the limited context window. The Gemini Gem must be trained to discern between consistently relevant information (Plot Essentials) and conditionally relevant information (Story Cards), advising creators on the most efficient placement of details to maximize narrative consistency and minimize token waste.

2.4 Author's Note

The **Author's Note** is a specialized component designed to provide the AI with specific guidance on the story's genre, preferred writing style, or overall tone. It acts as a direct influence on the AI's output generation, particularly for the immediate next response.
Unlike other narrative elements, the text within the Author's Note is sent to the AI towards the end of every player input. This strategic placement means it has a more direct and immediate impact on how the AI generates its subsequent output, influencing the "feel" of the current scene.
A unique characteristic of the Author's Note is its special formatting: the text is enclosed in square brackets [ ] when transmitted to the AI. This formatting is interpreted by the AI as descriptive indicators or instructions for what should come next, rather than as part of the actual story content. This interpretation stems from how the AI was trained, where bracketed text often served as authorial commentary or meta-instructions within the training data. This allows the Author's Note to be more technical and "out-of-character," functioning as explicit directives for the AI's generative process.
While optional, the Author's Note can significantly enhance narrative coherence and consistency. However, its brevity is crucial. Recommendations suggest keeping it short, ideally no more than 3 or 4 sentences, focusing on a couple of key instructions about the story's theme and setting. Excessive length can be detrimental, as a too-long Author's Note, due to its insertion point, can disrupt the flow of the most recent two actions and cause the AI to over-focus on its contents rather than the immediate narrative.
The relationship between Author's Note and AI Instructions is complementary. AI Instructions provide general, overarching directions and are placed at the beginning of the context. Author's Note, positioned near the bottom of the context, offers more direct and immediate guidance. This functional distinction implies that most general issues or broad stylistic preferences should be addressed in AI Instructions, while Author's Note should be reserved for fine-tuning specific scene instructions, setting guidelines, or theme reminders that require a more immediate influence on the AI's output. The Gemini Gem should be trained to advise creators on this hierarchical application of guidance, ensuring that the Author's Note is used judiciously for precise, short-term narrative steering, complementing the broader directives set by AI Instructions.

2.5 Story Cards

Story Cards are a dynamic feature designed to provide the AI with conditional memory, allowing for the inclusion of detailed information about specific story elements only when they become relevant. This mechanism helps manage the AI's limited context window by preventing constantly irrelevant information from consuming valuable tokens.

Each Story Card comprises several key components:

- **Type:** Primarily relevant for Character Creator Scenarios, influencing character selection. Otherwise, the AI generally ignores this field unless it's generating new Story Cards.
- **Name:** For the creator's reference only; the AI does not process this field.
- **Entry:** This is the core information transmitted to the AI when the card is activated. It is recommended to use plain English, concise language, and to place the most important details at the beginning and end, as the AI exhibits a bias towards information in these positions. The name of the subject should be repeated within the Entry, as the AI does not see the "Name" field.
- **Triggers:** These are keywords or phrases that, when detected in the AI's output or the player's input, activate the Story Card, causing its Entry information to be added to the context. Triggers are case-insensitive but sensitive to leading and trailing spaces.
- **Notes:** Ignored by the AI during adventures, this field is primarily used for player-facing descriptions in Character Creator Scenarios or for logging AI-generated content.

When a trigger word or phrase appears, the corresponding Story Card's Entry is prefaced with "World Lore:" and added to the context. Story Cards remain in context for a variable period after activation, depending on the context window size.

A critical operational nuance is that the AI cannot access Story Card information in the *same* output turn that the trigger word is generated. If the AI mentions a trigger, it might initially "wing it" and generate details inconsistent with the card's content. The Story Card's information only becomes available to the AI in the *next* turn. This means that if the AI makes an incorrect assumption about a character or element it just introduced, the player may need to correct the output or delete text after the trigger to allow the AI to properly load and utilize the Story Card on the subsequent turn.

For effective trigger design, creators should separate triggers by commas, generally without spaces (e.g., Amanda,your daughter). Caution is advised with short triggers that might be part of common words (e.g., cat could trigger on catalog). Strategic use of spaces around triggers (e.g., , troll , vs. ,troll,) can prevent unintended activations from words like "patrolling". Truncated words can also be used to catch singular and plural forms (e.g., boat for boats).

Story Cards are classified as "Dynamic Elements" in the AI's context assembly. They fill approximately 25% of the remaining tokens after "Required Elements" (like AI Instructions and Plot Essentials) have been accounted for. Their inclusion is prioritized based on the recency and frequency of their triggers. This dynamic loading contrasts sharply with Plot Essentials, which are *always* loaded. This distinction is vital for efficient context management; Plot Essentials are for information that is *always* relevant, while Story Cards are for details that are *conditionally* relevant. For example, a main character's core personality should be in Plot Essentials, but detailed lore about a specific, rarely encountered monster is better suited for a Story Card.

The Gemini Gem, when assisting with scenario creation, must be trained to understand this conditional loading mechanism and its implications for narrative flow. It should guide creators to design triggers that activate only when truly needed, optimizing token usage and preventing the AI from being overwhelmed by extraneous information. The Gem should also advise on the potential for AI inconsistencies during the *same turn* a trigger is activated, suggesting strategies for player intervention or more robust initial context setup in Plot Essentials for frequently appearing elements.

2.6 Triggers

Triggers are the specific keywords or phrases that activate Story Cards, causing their associated Entry information to be loaded into the AI's context. They are the critical link between the ongoing narrative and the vast, conditional memory stored in Story Cards.

The effectiveness of triggers lies in their precise design. They are case-insensitive, meaning

"Dragon" and "dragon" will both activate the same trigger. However, they are sensitive to leading and trailing spaces. This nuance is crucial; a trigger defined as "dragon" (with no spaces) would activate if "dragon" appears within a larger word like "dragonfly," which is usually undesirable. Conversely, a trigger defined as " dragon " (with leading and trailing spaces) would only activate when "dragon" appears as a standalone word. Creators must be strategic in their use of spaces to prevent unintended activations, especially with common words that might be part of other words (e.g., "troll" vs. "patrolling").
Multiple triggers for a single Story Card should be separated by commas, generally without spaces (e.g., Amanda,your daughter). Truncated words can be used to catch both singular and plural forms (e.g., boat will trigger on boats). The safest trigger types are typically proper names, as they are less likely to inadvertently appear in unrelated contexts.
Triggers are scanned from both the player's input and the AI's output. This means if the AI generates a word that is a trigger, the corresponding Story Card will be activated for the *next* turn. This delay is a significant operational detail: the AI cannot "stop mid-output to check what it's writing for triggers". Consequently, if the AI introduces a new character for whom a Story Card exists, it will generate initial details about that character *without* the benefit of the Story Card's information. Only on the subsequent turn will the Story Card be loaded into context, allowing the AI to incorporate its details. This can lead to temporary inconsistencies if the AI's initial "winging it" contradicts the Story Card's established lore.
The Gemini Gem, when advising on trigger creation, must emphasize the importance of precision in trigger definition, including the strategic use of spaces to prevent false positives. It should also educate creators about the one-turn delay in Story Card activation, preparing them for potential initial inconsistencies and suggesting ways to mitigate them, such as placing crucial, always-present character details in Plot Essentials rather than relying solely on Story Cards for initial introductions.

3. How AI Models Work in AI Dungeon

AI Dungeon is fundamentally powered by Large Language Models (LLMs) that function as highly advanced predictive text bots. These models generate narrative responses by predicting the most probable next "tokens" (words or sub-word units) based on the input context they receive. The quality and coherence of the generated narrative are directly tied to the information contained within this context and the various parameters that control the AI's generation process.

3.1 Context Window and Information Prioritization

The **Context Window** is the finite block of text that is fed to the AI model for it to generate its output. The AI essentially continues the narrative based on everything within this window, including not just the content but also the writing style and tone. Information appearing later in the context is generally weighted more heavily by the AI, as this mirrors how humans process sentences and paragraphs.
The context in AI Dungeon is comprised of two broad categories: **Required Elements** and **Dynamic Elements**.
Required Elements are generally included in their full length, up to a certain token limit. These include:

- **AI Instructions:** Placed at the very beginning of the assembled context, these provide foundational rules and global directives for the AI.
- **Plot Essentials:** Positioned after AI Instructions, these contain core, always-relevant story details like character traits and overarching plot points.

- **Story Summary:** A summary of the broader story details, working with the new Memory System.
- **Front Memory and Last Action:** These are always included in full, representing the most recent player input and AI output.
- **Author's Note:** Inserted near the bottom of the context, just before the most recent action, giving it a direct influence on the immediate output due to its proximity to the generation point.

If the combined length of Required Elements exceeds 70% of the total context size, a prioritization system trims or excludes less important sections to fit the limit. Front Memory and Last Action are always prioritized, followed by Author's Note, Plot Essentials, AI Instructions, and Story Summary.

Dynamic Elements are more flexible in their inclusion and fill the remaining tokens after Required Elements. These include:

- **Story Cards:** These constitute approximately 25% of the remaining token space. They are included based on the recency and frequency of their triggers within the last 4-9 actions.
- **History:** This comprises the recent story history, typically filling about 50% of the remaining tokens (up to 75% if Memory Bank is disabled).
- **Memory Bank:** This utilizes the remaining tokens (around 25%) and retrieves memories based on their relevance to the most recent action.

The final order of context elements sent to the AI model is: Instructions, Plot Essentials, Story Cards, Story Summary, Memory Bank, History, Author's Note, Last Action, and Front Memory. This specific ordering is crucial, as the AI's processing is influenced by the sequence of information.

The concept of "tokens" is fundamental to understanding LLM operation. Tokens are the units of text (words, sub-words, or characters) that the AI processes. The **Context Length** setting determines the maximum number of tokens that can be sent to the AI model per turn. Maximizing this setting, especially for models like Mixtral, MythoMax, Tiefighter, and GPT-4 Turbo that support larger contexts, is generally recommended to provide the AI with as much information as possible for coherent outputs. However, context length is often tied to membership tiers and can incur credit costs for higher values.

A significant observation is that relying excessively on "under the hood" features like World Info (which functions similarly to Story Cards) or Pinned Info (similar to Plot Essentials) for *all* details can be counterproductive. While these features are designed to provide persistent memory, over-reliance can "pollute the context" with brief, simple sentences that may inspire the AI to generate similarly basic outputs, rather than rich, evocative prose. Furthermore, it consumes context space that could otherwise be used for the immediate scene, potentially causing the AI to "forget" recent events more quickly. This suggests a delicate balance: use these components for crucial, high-level information, but allow the narrative flow to naturally develop other details. The Gemini Gem should guide creators to prioritize quality and relevance over sheer quantity in these "under the hood" components, optimizing for the AI's narrative generation capabilities rather than simply dumping information.

3.2 Advanced AI Settings

AI Dungeon provides several advanced settings that allow users to fine-tune the AI's generative process, influencing its creativity, consistency, and verbosity. These settings manipulate the mathematical calculations involved in the AI's probabilistic token selection.

- **Response Length:** This setting controls the maximum number of tokens the AI will output in a single turn. It is a matter of personal preference, allowing for short, quick responses or longer, more detailed ones. It is important to note that the set Response Length itself

consumes tokens from the overall context, meaning a higher response length reduces the tokens available for other contextual elements.

- **Temperature:** This parameter directly influences the randomness and creativity of the AI's responses. A higher temperature (closer to 1.0 or higher) increases unpredictability, leading to more varied and unique outputs, which can be beneficial for exploring unexpected storylines. Conversely, a lower temperature (closer to 0.0) results in more consistent and predictable responses, ideal for maintaining a coherent and focused narrative. The default is often 0.8, with suggestions to lower it for more reasonable responses or raise it for more uncommon text.
- **Top-K:** Available for some models, Top-K limits the AI's token choices to the 'K' most probable tokens for its response. By narrowing the possibilities, Top-K helps maintain relevance and consistency, preventing the AI from veering into irrelevant tangents. For example, a Top-K of 20 means the AI only chooses from the 20 most likely next tokens.
- **Top-P:** This setting filters out less likely tokens by establishing a probability threshold (e.g., 90%) and selecting the most likely tokens until their combined probability reaches this limit. This keeps the AI's outputs focused and on-topic, reducing randomness without making the output overly predictable. If Top-P is too low, outputs can become repetitive; if too high, they might be disjointed. A range of 0.5 to 0.95 is generally recommended.
- **Repetition Penalty:** Also available for some models, this setting adjusts probability values to make the AI less likely to repeat tokens. While seemingly beneficial, setting this value too high can penalize common words like "You," "I," "and," or even character names, potentially leading to grammatically incoherent or unusual outputs. Default is zero, and values significantly above 1 can lead to strange text.
- **Presence Penalty & Count Penalty:** These settings, available for some models, apply biases to tokens based on their prior appearance. Presence Penalty applies a fixed bias to tokens that have appeared at least once, while Count Penalty applies a bias proportional to the number of times each token has appeared. Frequency Penalty is similar to Count Penalty but is divided by the total number of tokens.

Experimentation is key to understanding the impact of these settings, as there are no universally "best values". The Gemini Gem should be designed to provide guidance on these advanced settings, explaining their effects on narrative generation and assisting creators in finding the optimal configuration for their desired story experience. This involves understanding how different settings influence the AI's creative latitude versus its adherence to established patterns.

3.3 Underlying AI Models

AI Dungeon utilizes a variety of Large Language Models (LLMs) to power its interactive storytelling, each possessing unique characteristics and specialties. These models range in size and architecture, from smaller models like Muse (12B) and Wayfarer Small (12B) to much larger and more powerful ones such as Wayfarer Large (70B), Mistral Small 3 (24B), Hermes 3 70B, DeepSeek V3 (671B), WizardLM 8x22B|39B, Hermes 3 405B, and Mistral Large 2 (123B). Some models, like Wayfarer Large, are specifically fine-tuned for "danger and drama," trained on datasets that emphasize conflict and unexpected outcomes. Others, like Mistral Small 3, are noted for their reasoning capabilities and consistency. Dynamic Large models offer a unique experience by switching between powerful AIs to keep the story fresh and unpredictable.
The choice of AI model directly impacts the context length available to the player, which varies by membership tier. For instance, Wayfarer Large offers context lengths from 2k (Adventurer tier) up to 16k (Mythic & above), while Mistral Small 3 can reach 32k tokens for Mythic tiers. More powerful models like GPT-4 Turbo can support even larger contexts, up to 128k tokens, though these often come with additional credit costs.

The AI's ability to maintain narrative coherence and consistent characterization is a key challenge for LLMs, especially in long narratives. LLMs may struggle with tracking evolving character relationships, inferring causal links between events, and maintaining a "Theory of Mind" for characters' motivations and emotional states. Human-written stories often exhibit more suspense, arousal, and diverse narrative structures compared to LLM-generated narratives, which can sometimes be homogeneously positive or lack tension. LLMs may also introduce plot holes or repetitive themes.

To enhance narrative generation, research suggests that explicit integration of discourse features like story arcs and turning points can improve suspense, emotion provocation, and narrative diversity. AI Dungeon's various components, such as Plot Essentials and Story Cards, are designed to address these inherent LLM limitations by providing structured context that helps the AI maintain consistency and depth.

The Gemini Gem, when assisting with scenario creation, should possess an understanding of the strengths and weaknesses of different AI models available in AI Dungeon. It should be able to recommend specific models based on the desired narrative style, conflict level, and complexity, and advise on how to best leverage scenario components to mitigate common LLM challenges, such as repetition or character inconsistency. This includes guiding creators on how to structure their input to encourage more complex story arcs and character development, compensating for areas where LLMs naturally fall short.

4. Conclusions and Recommendations for Gemini Gem Development

The detailed analysis of AI Dungeon scenario components and their interaction with underlying LLMs reveals a complex ecosystem where effective narrative generation hinges on strategic design and nuanced understanding of AI mechanics. For the development of a Gemini Gem aimed at assisting AI Dungeon scenario creation, several key conclusions and actionable recommendations emerge:

1. **Differentiate Between Human-Facing and AI-Facing Components:** The Gemini Gem must be trained to recognize and prioritize the distinct audiences for different scenario elements. For components like the Description, the Gem should optimize for human readability, discoverability, and clear player instructions. Conversely, for Prompt, AI Instructions, Plot Essentials, Author's Note, and Story Cards, the Gem's focus must shift to precise, AI-interpretable language and structure, adhering strictly to LLM operational nuances. This requires the Gem to have distinct generative modes or sub-personas.
2. **Understand the "Ephemeral Foundation" of the Prompt:** The initial Prompt serves as a powerful narrative catalyst, but its direct influence on the AI's ongoing generation diminishes rapidly due to the limited context window. The Gem should guide creators to leverage the Prompt for setting the immediate scene, character's starting state, and inciting incident. For persistent narrative elements, the Gem must advise strategic placement within Plot Essentials. This requires the Gem to model the "decay" of information within the LLM's context.
3. **Prioritize Positive and Specific AI Directives:** The AI responds more effectively to positive instructions (what *to do*) rather than negative ones (what *not to do*). The Gemini Gem should be trained to generate AI Instructions and Plot Essentials using clear, affirmative phrasing to steer the narrative. It should also emphasize specificity in descriptions and action verbs in prompts, as these directly enhance the AI's ability to generate vivid and relevant responses.
4. **Master Context Management Hierarchy:** The Gem needs a deep understanding of how information is prioritized and assembled in the AI's context window. It should guide

creators on the distinct roles of AI Instructions (global, foundational directives), Plot Essentials (always-loaded, core memory), Story Cards (conditionally loaded, detailed lore), and Author's Note (immediate, stylistic adjustments). The Gem should advise on the optimal placement of information to maximize narrative consistency and minimize token waste, recognizing that Plot Essentials are for constant relevance while Story Cards are for conditional relevance.

5. **Account for Dynamic Element Latency:** The Gemini Gem must educate creators about the one-turn delay in Story Card activation. When a trigger is generated by the AI, the corresponding Story Card content is not available until the subsequent turn. The Gem should provide strategies to mitigate potential inconsistencies during this delay, such as advising on initial character or setting details within Plot Essentials for frequently appearing elements.
6. **Optimize for Token Economy:** The limited token capacity of the AI's context window is a fundamental constraint. The Gem should guide creators in crafting concise yet dense Plot Essentials and Story Card Entries to maximize the information conveyed within the token limits. It should also highlight how Response Length settings consume tokens from the overall context, influencing the available space for other narrative components.
7. **Leverage Advanced AI Settings:** The Gem should explain the impact of Temperature, Top-K, Top-P, and Repetition Penalty on narrative generation. It should assist creators in experimenting with these settings to fine-tune the AI's creativity, consistency, and verbosity according to their desired storytelling style.
8. **Adapt to AI Model Characteristics:** Different AI models within AI Dungeon possess unique strengths and weaknesses. The Gemini Gem should incorporate knowledge of these model variations, recommending specific models based on the desired narrative style (e.g., drama, reasoning, creative prose) and advising on how to best utilize scenario components to compensate for inherent LLM challenges like repetition or character inconsistency.

By integrating these principles into its training, the Gemini Gem can become an invaluable tool for AI Dungeon scenario creators, enabling them to construct more coherent, engaging, and precisely controlled interactive narratives.

Works cited

1. Getting Started - AI Dungeon Guidebook, https://help.aidungeon.com/getting-started 2. AI Dungeon, https://aidungeon.com/gauntlet 3. What are Scenarios? - AI Dungeon Guidebook, https://help.aidungeon.com/faq/what-are-scenarios 4. Notes on how the context / memory works, and tips to possibly help the AI remember things : r/AIDungeon - Reddit, https://www.reddit.com/r/AIDungeon/comments/mm1ylz/notes_on_how_the_context_memory_works_and_tips_to/ 5. Some Advice for new player in Ai DUngeon : r/AIDungeon - Reddit, https://www.reddit.com/r/AIDungeon/comments/1503sys/some_advice_for_new_player_in_ai_dungeon/ 6. What is Plot Essentials? - AI Dungeon Guidebook, https://help.aidungeon.com/faq/plot-essentials 7. Best Practices : r/AIDungeon - Reddit, https://www.reddit.com/r/AIDungeon/comments/k5op6o/best_practices/ 8. What are Story Cards? - AI Dungeon Guidebook, https://help.aidungeon.com/faq/story-cards 9. What is Advanced AI Settings? - AI Dungeon Guidebook, https://help.aidungeon.com/faq/what-are-advanced-settings 10. What goes into the Context sent to the AI? - AI Dungeon Guidebook, https://help.aidungeon.com/faq/what-goes-into-the-context-sent-to-the-ai 11. Prompting Techniques | Prompt Engineering Guide, https://www.promptingguide.ai/techniques 12. Mastering Prompt Engineering: A Developer's Guide to Harnessing AI Effectively - Medium, https://medium.com/@williamwarley/mastering-prompt-engineering-a-developers-guide-to-harne

ssing-ai-effectively-923c3f71a484 13. What is AI Instructions? - AI Dungeon Guidebook, https://help.aidungeon.com/faq/ai-instructions 14. Authors Note : r/AIDungeon - Reddit, https://www.reddit.com/r/AIDungeon/comments/1dxj5x3/authors_note/ 15. What is Author's Note? - AI Dungeon Guidebook, https://help.aidungeon.com/faq/what-is-the-authors-note 16. What's the difference between story cards and plot essentials? : r/AIDungeon - Reddit, https://www.reddit.com/r/AIDungeon/comments/1f6nltd/whats_the_difference_between_story_cards_and_plot/ 17. Do story card keywords trigger when being used by player specifically or do mentions of them in the output work too? And are keywords case sensitive? : r/AIDungeon - Reddit, https://www.reddit.com/r/AIDungeon/comments/1juuio2/do_story_card_keywords_trigger_when_being_used_by/ 18. Story card : r/AIDungeon - Reddit, https://www.reddit.com/r/AIDungeon/comments/1b73289/story_card/ 19. Absolute Beginner's Guide To AI Dungeon - YouTube, https://www.youtube.com/watch?v=J0nxtUYsX7U&pp=0gcJCdgAo7VqN5tD 20. The Full Context on Using Credits for Context : r/AIDungeon - Reddit, https://www.reddit.com/r/AIDungeon/comments/1ks3cy2/the_full_context_on_using_credits_for_context/ 21. AI Model Differences - AI Dungeon Guidebook, https://help.aidungeon.com/ai-models-and-their-differences 22. Narrative Understanding with Large Language Models | The Alan Turing Institute, https://www.turing.ac.uk/work-turing/research-and-funding-calls/ai-fellowships/yulan-he-project 23. Are Large Language Models Capable of Generating Human-Level Narratives? - ACL Anthology, https://aclanthology.org/2024.emnlp-main.978.pdf 24. Any tips to make AI Dungeon more enjoyable? : r/AIDungeon - Reddit, https://www.reddit.com/r/AIDungeon/comments/1l063ti/any_tips_to_make_ai_dungeon_more_enjoyable/ 25. Ember - AI Dungeon, https://aidungeon.com/ember 26. Best Roleplay AI Chatbots 2025: Free - Uncensored Options for Creative Storytelling, https://screenapp.io/blog/best-roleplay-ai-chatbots/