

Other Business Applications: Game Design, Financial Analysis etc.

Contents

1. LLM-based Narrative Games
2. Converging Narratives
3. LLMs in Finance
4. Further Applications
5. References

LLM-based Narrative Games

- Traditional games are built in a form of a predefined storyline that should be followed by the player: there are key tasks, key events (triggers), and usually a predetermined narrative plot [1]
- Game designers utilize *narrative branching* for more entertaining and immersive experience: that is **interactive** storytelling where player choices create divergent story paths, sometimes allowing multiple outcomes and personalized experiences that enhance **player agency**: player's meaningful choices do affect the final state of the game. That holds both for events and dialogues [1, 2, 4]
- Branching narratives make games feel more responsive and alive, and different storylines make it interesting for the player to restart the game to **explore** different paths/endings
- The branches may be completely independent, or may share key events, may come and not come to the same ending(s) – all that is up to the game designers

LLM-based Narrative Games

- Rule-based and scripted approaches to narrative branching are too rigid—developers must anticipate every possible player action and pre-write all dialogue/event trees, severely limiting creative possibilities => it is practically impossible to create truly open-ended, scenario-free narratives that can adapt to **unexpected** player behavior [1, 2, 3]
- Solution: utilize LLMs to generate contextually appropriate responses to any player input, breaking free from pre-scripted limitations and enabling dynamic storytelling that is being generated on the fly. Strong capabilities of LLMs to create structured output and call tools can enable not only dynamic dialogue generation but also translation of natural language commands into actions performed by the agents in the game
- LLMs enable emergent narratives where stories **evolve organically** based on player interactions, creating unique experiences that even developers cannot fully predict

LLM-based Narrative Games

- The simplest approach to integrating LLM capabilities into the gaming is to create a **structured foundation** with defined settings, characters, locations, key plot points, and objects/clues that serve as the narrative backbone while allowing dynamic content generation [1, 6]
 - Each NPC receives system prompts defining their role, traits, secrets, relationships, and behavioral constraints — creating consistent yet responsive character interactions
 - Dynamic state management: the game maintains (usually JSON-based) state to track discovered locations, clues, character relationships, completed milestones, and conversation history to ensure narrative coherence across interactions
 - Player input is processed by combining current game state with relevant prompts, then queried to the LLM for contextually appropriate dialogue or narrative events
 - Some systems automatically convert gameplay logs into node-graphs, helping developers visualize and optimize emergent narrative paths (allowing for a more informed improvements in later patches and releases)
 - Such games end through milestone-based triggers (collecting required clues, confronting antagonists) or emergent endpoints detected via keyword analysis
 - Filtering systems prevent contradictions to established facts; dynamic hint injection guides struggling players toward narrative progression without breaking immersion

LLM-based Narrative Games

Setting Prompt

You are the storyteller in the following setting: you are in the Arcane Academy, a magical school where the Grand Library has been cursed. Key locations: Library, Alchemy Lab, Headmaster's Office. Key objects: Ancient Tome, Crystal Orb, Enchanted Quill

Characters

Librarian Elara

Role: Head Librarian

Personality: Nervous, protective of books

Secret: Knows the curse origin but fears speaking

Relationship: Distrusts the Alchemy Professor, respects the Headmaster

Alchemy Professor

...

Headmaster

...

Game State

```
{
  "clues": [],
  "relationships": {
    "elara_trust": 0.5
  },
  "location": "lib_entrance",
  "milestones": []
}
```

Player Input

Ask Librarian Elara about the strange whispers coming from the restricted section

Assemble Prompt

[Setting Prompt] + [Elara Character Prompt] +
Game State: No clues found yet, player at library entrance +
Player Action: Ask about whispers

LLM Output

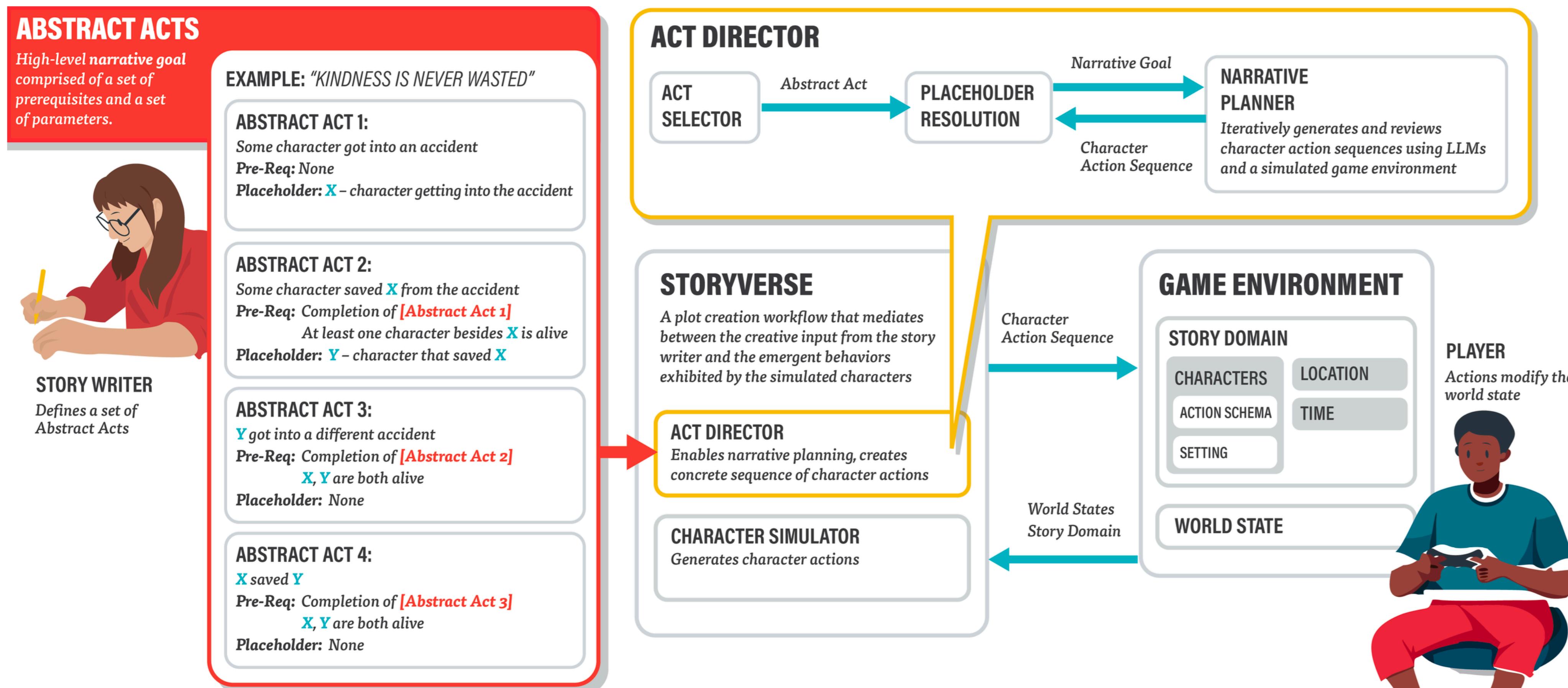
Elara's eyes dart nervously toward the restricted section. 'The whispers... they started three nights ago when Professor Grimwald brought that ancient tome for research. I shouldn't say more, but... perhaps you could investigate the Alchemy Lab?'

Game State

```
{
  "clues": [
    "ancient_tome_hint"
  ],
  "relationships": {
    "elara_trust": 0.7
  },
  "location": "library",
  "milestones": [
    "first_clue_discovered"
  ]
}
```

LLM-based Narrative Games

- Systems like StoryVerse [3] use high-level „abstract acts“ that the LLM dynamically sequences and “instantiates”, creating emergent narratives that hold the balance between the players freedom and the initial authorial intent



LLM-based Narrative Games

STORY DOMAIN

CHARACTERS



Isabella Rodrigues is the owner of Hobbs Cafe and loves making people feel welcome.



Eddy Lin is a student at Oak Hill College studying music theory and composition.



Abigail Chen is a digital artist and animator who loves to explore how to use technology to express ideas.

LOCATIONS



ACTION SCHEMA

MoveTo(**X**)
Think(**X**)
TalkTo(**X, Y**)
Study(**X**)

Work(**X**)
ShopFor(**X**)
Eat(**X**)
Drink(**X**)

Sleep()
Cook(**X**)

(SELECT) CHARACTER SIMULATION



Thinks("I need to make sure everything is ready!")
Isabella needs to prepare for the Valentines Day party at **Hobbs Cafe**. She needs to plan ahead.

ACT 1 SOME CHARACTER GOT INTO AN ACCIDENT



MoveTo(**JohnsonPark**)
Johnson Park is quiet and inspiring, perfect for working on my music composition project.

Work(**Music**)

Eddy is deeply engrossed in his work, feeling the creative flow.

MoveTo(**Market**)

I need to pick up snacks and water to keep my energy up while I work!

ShopFor(**Snacks**)

Eddy selects snacks and a bottle of water.

MoveTo(**JohnsonPark**)

Back to **Johnson Park** to continue my composition.

ACT 2 SOME CHARACTER SAVED [EDDY] FROM THE ACCIDENT



MoveTo(**JohnsonPark**)
I'm going to the **park** for some inspiration for my project.

Think(**Inspired**)

Abigail feels optimistic about finding inspiration in the **park's** natural beauty.

ACT 3 [ABIGAIL] GOT INTO A DIFFERENT ACCIDENT



MoveTo(**SupplyStore**)
I need to pick up some art supplies for my project.

ShopFor(**Supplies**)

These supplies will help bring my animation project to life!

MoveTo(**JohnsonPark**)

I'll head back to the **park** to sketch some scenery.

ACT 4 [EDDY] SAVED [ABIGAIL]



Work(**Music**)
With the **park's** calm ambiance, **Eddy** finds the creativity flowing effortlessly.



TalkTo(**Eddy**, "Your music is so inspiring!")
Abigail finds solace in **Eddy's** music, feeling motivated to start her project anew.



TalkTo(**Abigail**, "Let's inspire each other!")
Eddy is grateful for **Abigail's** words and feels a sense of camaraderie

Think("I'm glad Abigail is here...")
Eddy sees potential in a collaborative project.



Think("That'd give my project a unique dimension!")
Abigail is excited about the possibility of incorporating **Eddy's** music into her animation.

(SELECT) CHARACTER SIMULATION



MoveTo(**HobbsCafe**)
A change in scenery might spark new ideas for my composition. Plus, a cup of coffee could really help me focus.

LLM-based Narrative Games

- Unlike framed systems, *adaptive games* start with minimal predefined structure (e.g. only a genre prompt) and let the LLM invent settings, characters, and plotlines **in real time** based on player interactions; thus, the entire game unfolds in an improvised manner [3, 5]
- Advanced systems can generate new quests, NPCs, and locations on-demand, responding to player choices with content that didn't exist until the moment of interaction [5]
- [3] uses LLMs to interpret natural language player commands and generate executable behavior trees, enabling fluid agent control beyond traditional menu systems
- Playing such games, players can issue commands in natural language ("Attack only if enemy is weak, otherwise heal"), which LLMs translate into structured behavior logic for the game agents to execute or the trigger to develop the story in a certain way [3, 7]
- The control is quite fluid in this setup: it is often possible to roll back to the previous step to regenerate the current one or just decide to not interfere with the game, e.g. [5]

Converging Narratives

- As LLM-generated narratives **branch freely**, stories risk becoming incoherent or fragmenting into isolated paths that never reconnect, potentially leaving players with unsatisfying or incomplete narrative experiences
- [4] demonstrates that given two versions of a story, individual LLMs struggle to generate a continuation that would fit both => evidence that **when not controlled** on purpose, branch divergence grows as a snowball as the story evolves
- [4] combines probability distributions from two narrative branch prompts, allowing LLMs to generate text that satisfies both story contexts and enables natural narrative rejoining. By consolidating token probabilities from different narrative prompts (e.g., "hero confronts villain directly" vs. "hero gathers allies for ambush"), the system produces semantically compatible continuations that merge divergent storylines
- [4] discovered that successful convergence requires prompts to share at least minimal **core** elements (characters, locations) – otherwise the approach would either omit the contradicting elements or be appropriate only for one of the prompts

LLMs in Finance

- Many tasks in finance require manual processing of large financial data for some decision making: news and charts for investments, bank operations for fraud detection and credit scoring etc.
- Since LLMs can process vast amounts of structured and unstructured data and excel and pattern discovery, they can take on this kind of data analysis
- => LLMs are used for the tasks named above as well as for customer service & advisory through chatbots, and automated report generation; furthermore, they are used for real-time market monitoring [8]
- Because of the specificity of the domain, fine-tuning on specific financial datasets or even training custom finance LLMs from scratch (like BloombergGPT) is quite customary [8, 9]
- Tool-augmented generation capability with external APIs and calculators often accompanies such models as there arises an increased **demand for accuracy** [8, 10]
- Example: the LLM can read the latest news and predict the best strategy for the user's relevant assets (sell / buy); or it can analyze hundreds of the latest news and recommend a portfolio provided a budget

LLMs in Finance

- Fine-tuned models improve understanding financial context, and domain-specific reasoning patterns that general-purpose LLMs miss, leading to more accurate and reliable financial insights [8]
- These specialized models consistently outperform both traditional financial analysis methods and general AI systems when evaluated on actual financial metrics [8, 9, 10]
- FinLlama [9] demonstrates how domain-adapted LLMs can generate **actionable trading signals** by analyzing financial news sentiment
 - Based on Llama-2-7B, fine-tuned on 34k+ labeled samples from four financial news datasets annotated as positive, negative, or neutral
 - Creates long-short portfolios with top 35% most positive companies held long and bottom 35% most negative held short using equally-weighted strategy after processing 200k financial news articles (2015–2021)
 - Achieved 308.2% cumulative returns, 45.0% annualized return
 - RAVEN [10] combines natural language understanding with precise computational tools like calculators and SQL engines for tabular data analysis

Further Applications

- There are **numerous industries** LLMs are actively integrated to besides the ones we have discussed in this and in the previous sessions
 - Healthcare & Pharmacy: clinical decision support, drug discovery
 - Legal: contract analysis, compliance monitoring
 - Scientific research (**up next!**): literature review, hypothesis generation, and summarization of scientific findings across domains
 - Electronic design automation: automated HDL (hardware description language) script generation, verification etc. [11]
 - Content generation: e.g. [12] automates research, outline creation, and section-by-section drafting for long-form, well-organized Wiki articles
 - Automated planning & scheduling: [13] solves complex planning tasks (e.g., for traveling) with formal verification and constraint satisfaction
 - Real-time analysis of sensor data for environmental or machine monitoring for automated report generation, anomaly detection
 - + any application gives a „placeholder“ to plug in a domain-specific chatbot
- => (Almost?) any scenario where **LLM-recognizable patterns** must be discovered for automation gives a room for LLM integration

References

- [1] [Player-Driven Emergence in LLM-Driven Game Narrative](#), Microsoft Research
- [2] [StoryVerse: Towards Co-authoring Dynamic Plot with LLM-based Character Simulation via Narrative Planning](#), Autodesk Research
- [3] [Game Agent Driven by Free-Form Text Command: Using LLM-based Code Generation and Behavior Branch](#), University of Tokyo
- [4] [Generating Converging Narratives for Games with Large Language Models](#), U.S. Army Research Laboratory
- [5] [AI Dungeon Games](#), AI Dungeon (game catalogue)
- [6] [AI Town](#), Andreessen Horowitz & Convex (game)
- [7] [gigax](#), GigaxGames (GitHub repo)
- [8] [Large Language Models in Finance: A Survey](#), Columbia & New York University
- [9] [FinLlama: Financial Sentiment Classification for Algorithmic Trading Applications](#), Imperial College London & MIT
- [10] [Equipping Language Models with Tool Use Capability for Tabular Data Analysis in Finance](#), Monash University
- [11] [LLM4EDA: Emerging Progress in Large Language Models for Electronic Design Automation](#), Shanghai Jiao Tong University et al.
- [12] [Assisting in Writing Wikipedia-like Articles From Scratch with Large Language Models](#), Stanford
- [13] [Large Language Models Can Solve Real-World Planning Rigorously with Formal Verification Tools](#), MIT, Harvard University & MIT-IBM Watson AI Lab