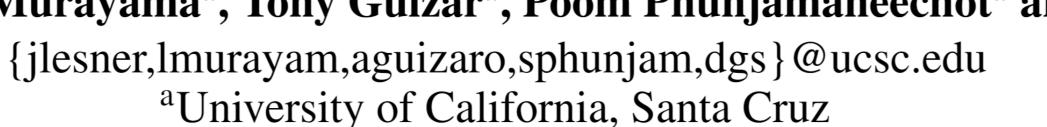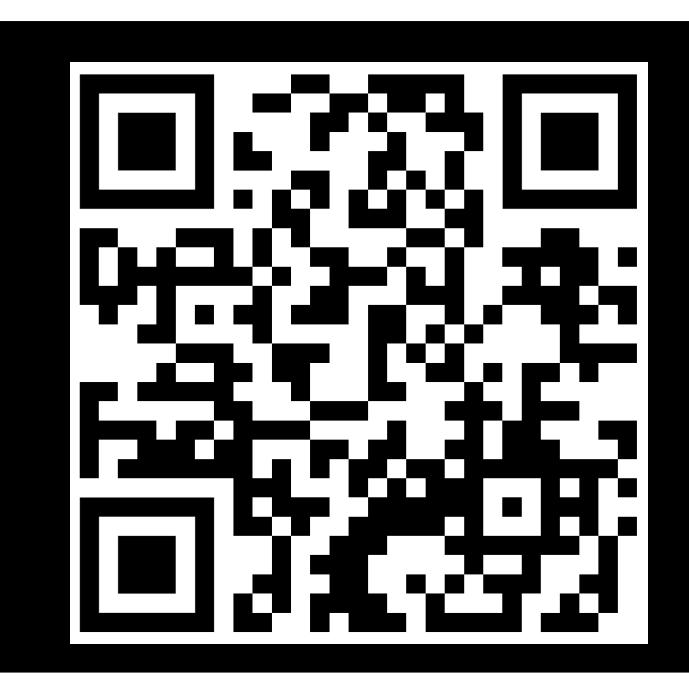# AI Personalized Interactive Fiction for Young Children
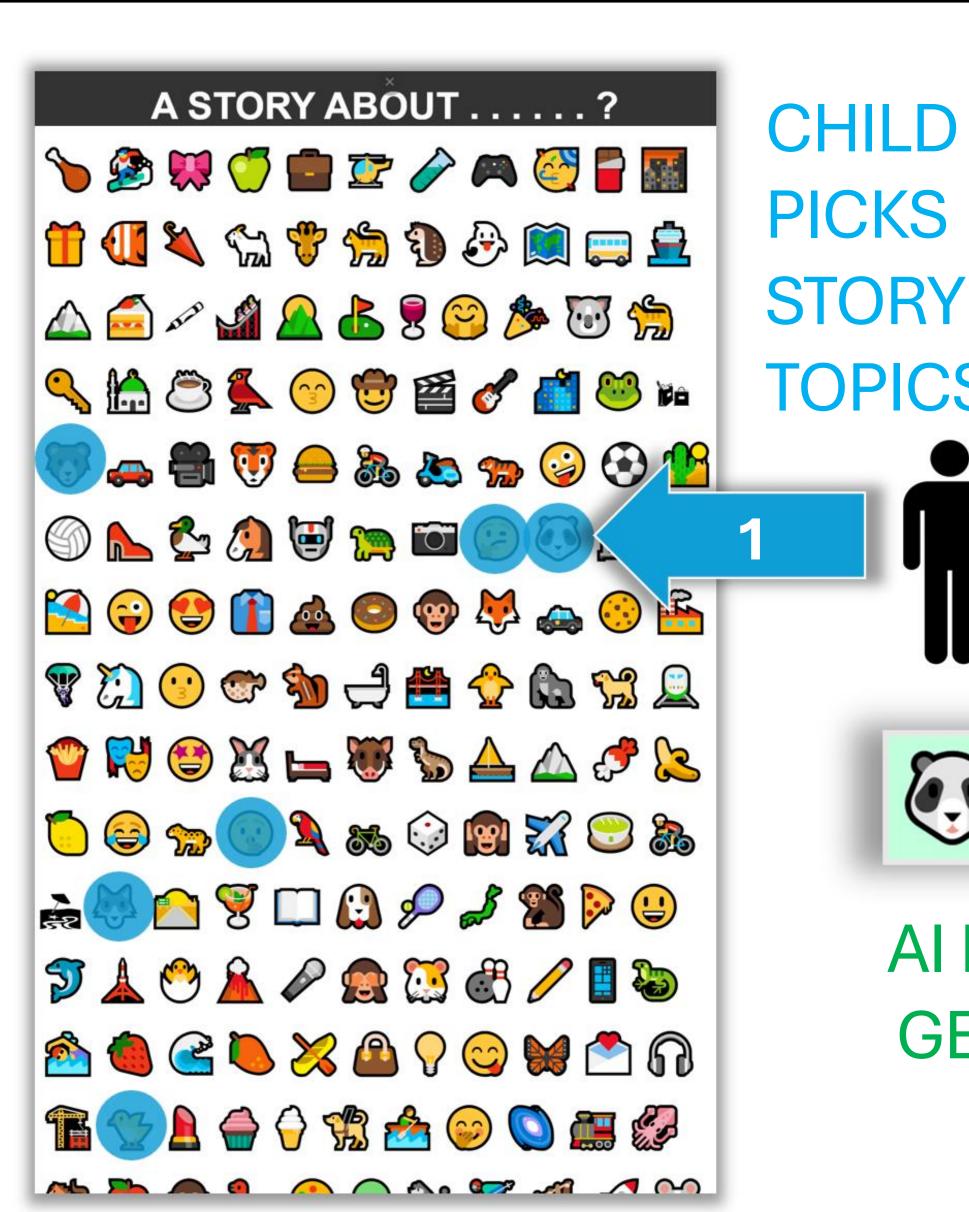
Jasmine Lesner[a], Luke Murayama[a], Tony Guizar[a], Poom Phunjamaneechot[a] and Daniel Shapiro[a]

{jlesner,lmurayam,aguizaro,sphunjam,dgs} @ucsc.edu

[a]University of California, Santa Cruz

ECAI 50th ANNIVERSARY

UC SANTA CRUZ — BaskinEngineering

Choose Your Own AI-venture: A Prototype for Young Readers
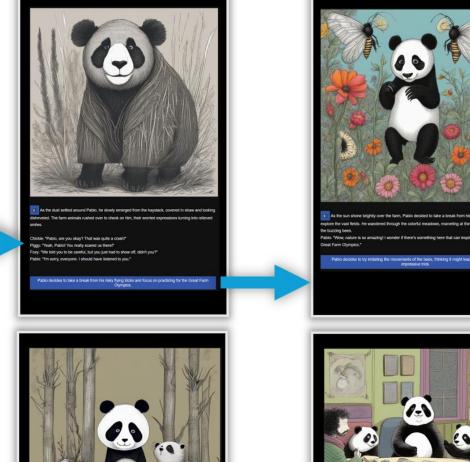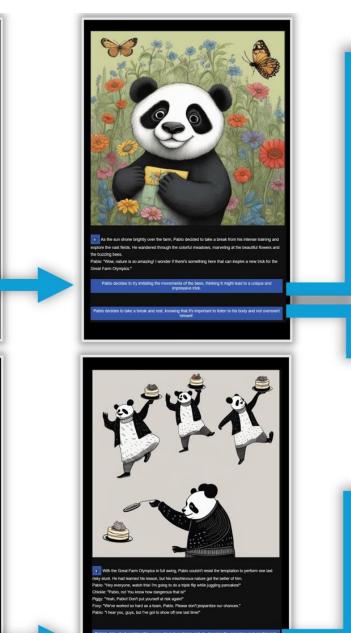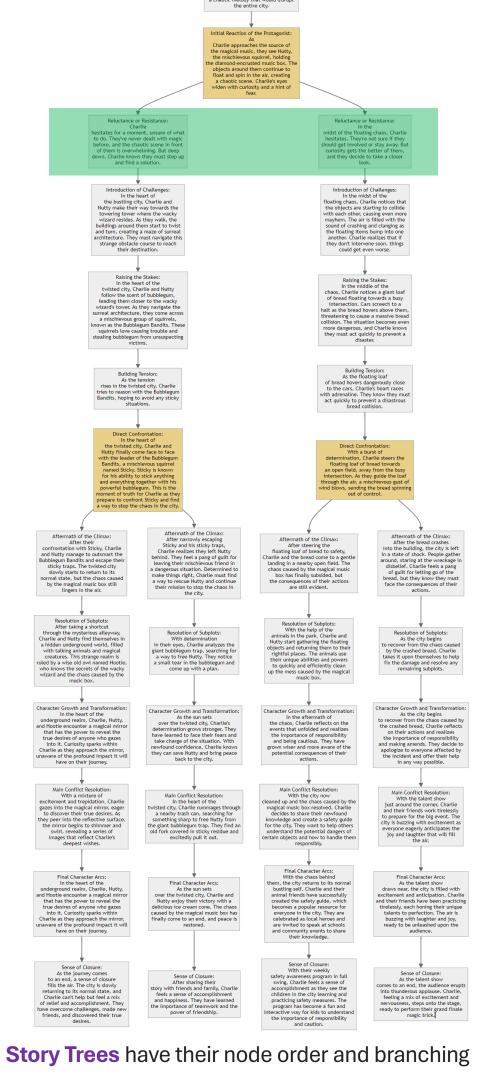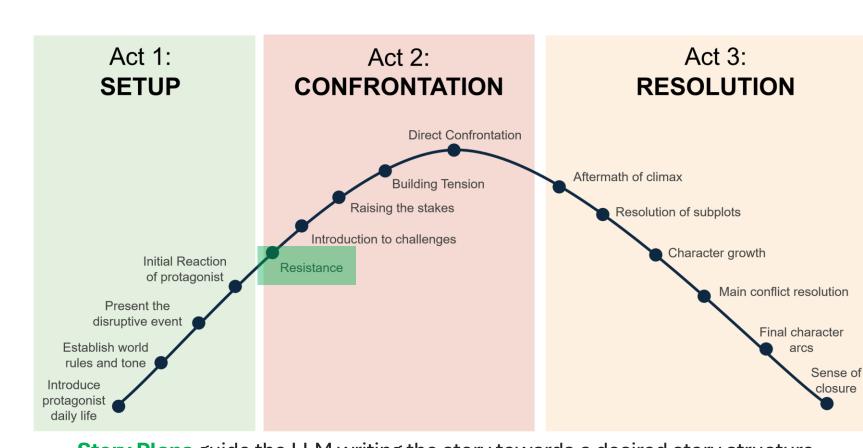

CHILD PICKS STORY TOPICS

1

AI MODELS GENERATE STORY
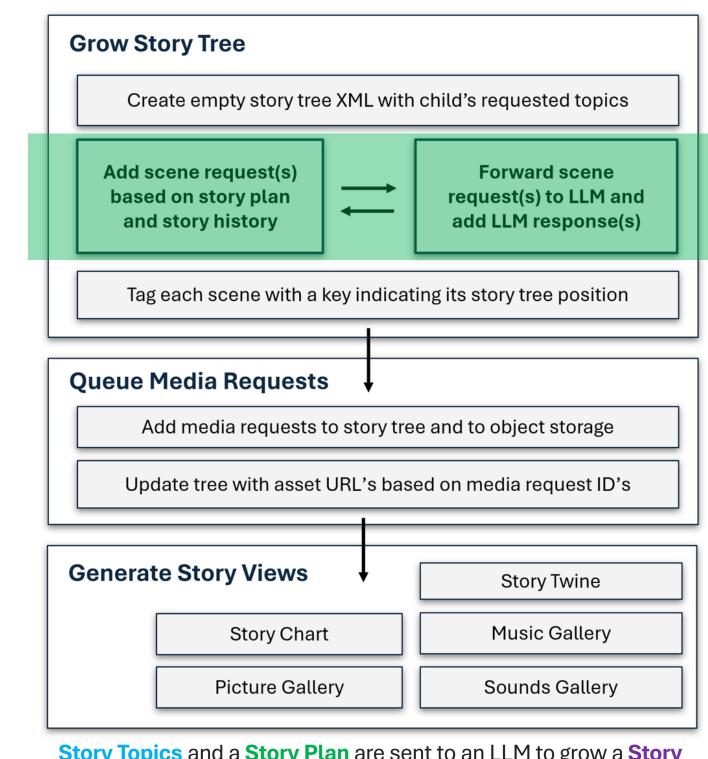
2

A story with pictures, sounds, music, and choices!

1. Child picks topics for their story.
2. AI models generate story and galleries.
3. Parent gives feedback using galleries.
4. AIs update story as guided by parent.
5. Child enjoys their personalized story.

5

1

2 2 3

4

Act 1: SETUP — Act 2: CONFRONTATION — Act 3: RESOLUTION

Story Plans guide the LLM writing the story towards a desired story structure.

Grow Story Tree
- Create empty story tree XML with child's requested topics
- Add scene request(s) based on story plan and story history ⇄ Forward scene request(s) to LLM and add LLM response(s)
- Tag each scene with a key indicating its story tree position

Queue Media Requests
- Add media requests to story tree and to object storage
- Update tree with asset URL's based on media request ID's

Generate Story Views
- Story Chart
- Story Twine
- Picture Gallery
- Music Gallery
- Sounds Gallery

Story Trees have their node order and branching controlled by a Story Plan. LLM prompts accumulate the "story so far" to grow each branch according to the selected Story Topics.

Story Topics and a Story Plan are sent to an LLM to grow a Story Tree containing story branches and descriptions of pictures, sounds and music. These descriptions are then queued as media requests to other AI models.

Each story scene has pictures, sounds and music. One or more large blue buttons advance the story to the next scene, depending on the text displayed on the button.

Galleries allow parents to efficiently review all pictures, sounds, and music. Clicking RETRY (or UPDATE) causes a media item to be re-attempted with the same (or an updated) AI model prompt.

Worker Pool

User Interface
- Story List View / Story Suggest View
- Story Page View
- Flow Chart View / Sound Gallery View
- Music Gallery View / Picture Gallery View
- story_list() / story_suggest() ... retry_request() / edit_request()

Object Storage Service
- User Interface Views
- User Assets: Twines Pictures Sounds Music
- Request Queue
- Request Logs

Web Service
- Web API / Storage Adapter

Story Worker
- Poll Storage for Story Request → GENERATE STORY → Publish Story Views / Queue Media Requests

Picture Worker
- Poll Storage for Picture Request → Generate Picture → Publish Picture to Storage

Music Worker
- Poll Storage for Music Request → Generate Music → Publish Music to Storage

Sound Worker
- Poll Storage for Sound Request → Generate Sound → Publish Sound to Storage

GPT-3.5 Turbo — OpenAI

Stable Diffusion XL

MusicGen 3.3B — AudioCraft Research By Meta AI

Suno.ai Bark

User Interface queues story suggestions and parental requests to update story pictures, sounds and music. Worker Pool handles requests in parallel using various AI models on available GPUs and APIs.