

Procedural Quest Generation

Max Quintavalle





Why?



Procedural Content Generation

Procedural generation is the idea of generating data algorithmically instead of manually

Procedural Content Generation is the process of generating that data for some kind of service that would be content for someone

Examples of this in video games would be minecraft with its world generation

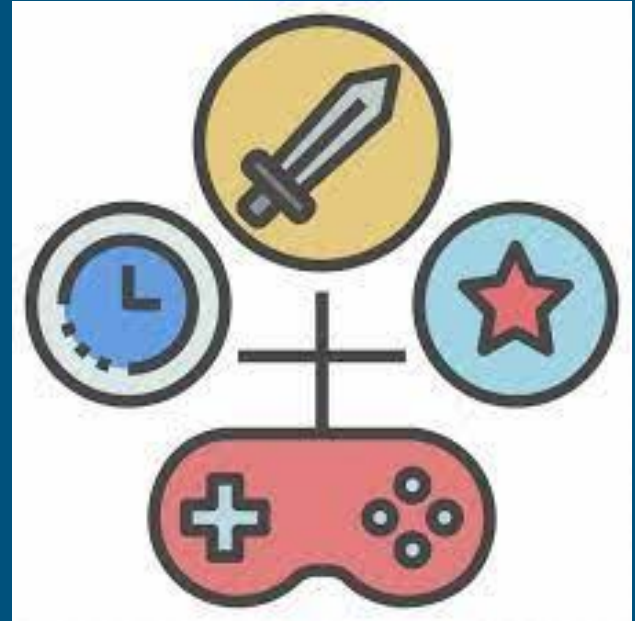
Procedural Quest Generation

Procedural Quest Generation is the idea to use procedurally generate content for quests that you would find in video games.

This can be done Completely through AI or through a mixed-initiative approach

Outline

- ❑ Background/Introduction
 - Large language models
 - Knowledge graphs
- ❑ Player input based PQG
 - Pipeline method
 - Studies 1 & 2
- ❑ Mixed Initiative PQG
 - QuestGram method
 - Studies 1 & 2
- ❑ Conclusion
- ❑ Questions

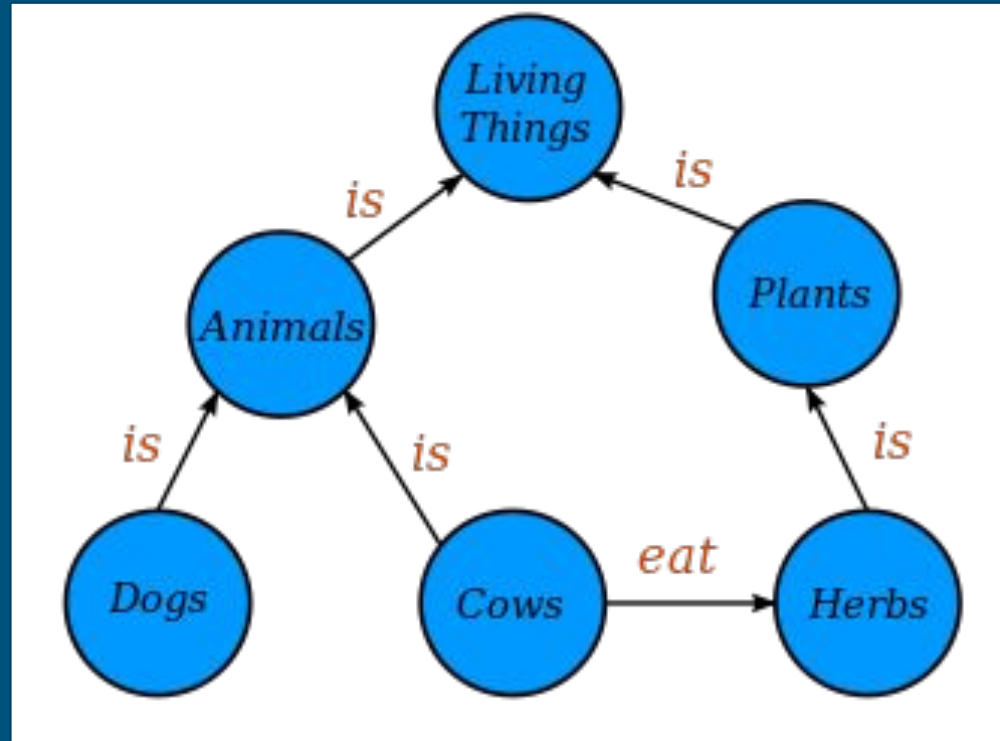


Knowledge Graphs

Knowledge graphs are made of nodes and edges:

Nodes - objects

Edges - relationships



Why do you need to know this



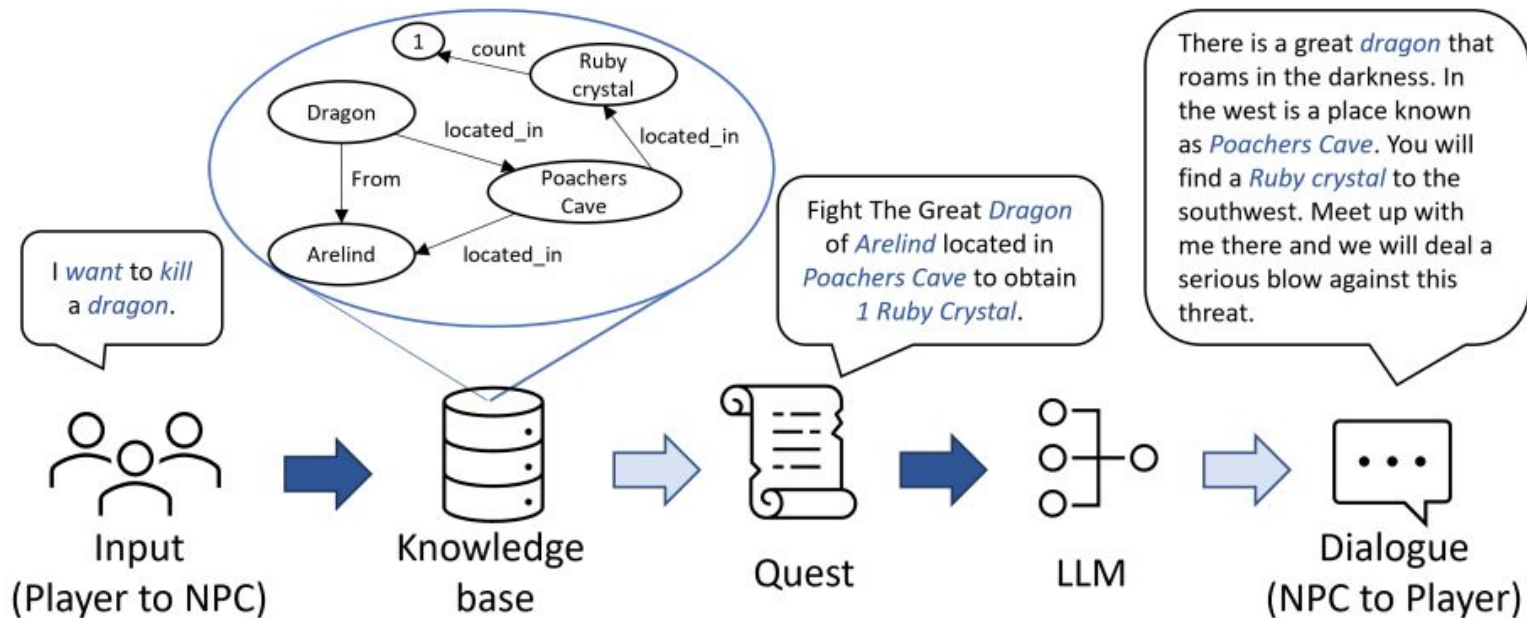
Pipeline Method

The purpose of the pipeline method is to create more individuality for the player experience and add replayability

They organize their data into tuples which are set in combinations of quest, title and dialogue

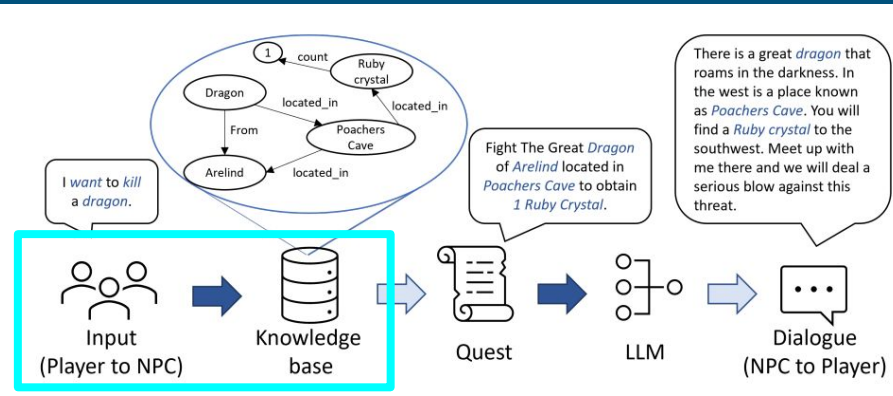
This process is still in research and hasn't actually been applied to any popular game

Pipeline Method



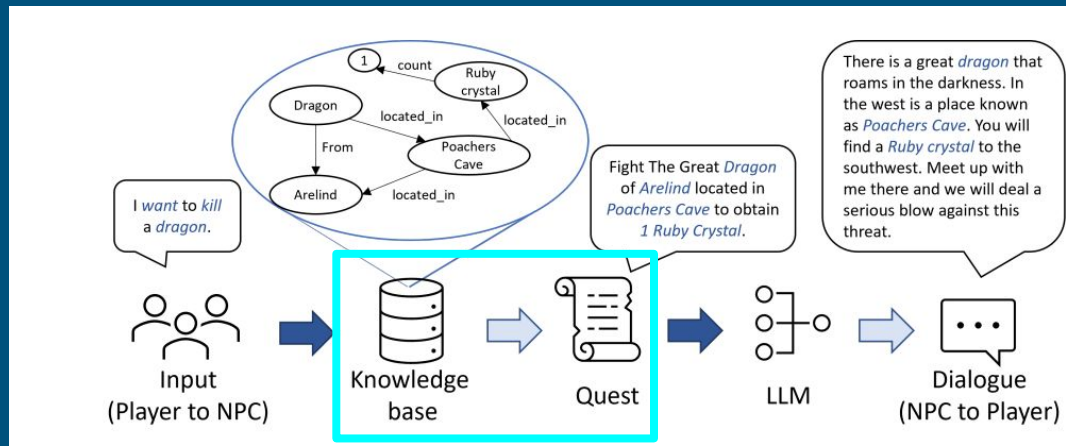
Pipeline Method(Knowledge Graph)

- ❑ Start with the NPC's node on the knowledge graph
- ❑ Then filter through the knowledge graph based on the cosine score and user input
- ❑ Usually has a traversal depth of 2



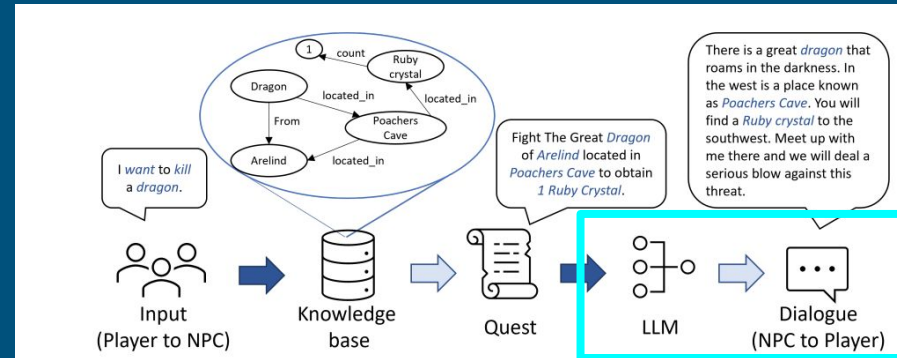
Pipeline Method(Quests)

- ❑ The player input is now classified into a type
 - Combat
 - Gathering
 - Exploration



Pipeline Method(LLM)

- ❑ The LLM used for the pipeline is called DRG-L which is trained on WOW(World of Warcraft) quests with all the data organized into tuples(quest, title, dialogue)
- ❑ The LLM is expected to receive a quest and make a title and dialogue based off of the quest.



Study 1 : Dialogue Quality

A Study was done on the level of quality of different language models and that of WOW

This study was done on 208 students from Brigham Young University by giving them a random assortment of 16 title and dialogue pairs.

DRG-L	VS-MYS	WOW
OpenAI's GPT-2 trained on the data set of WOW quests	GPT-2 model fined tuned by a different study by Stegeren and Myśliwiec but doesnt fit the tuple model	Hand crafted Quests made by the developers of WOW

Study 1 : Dialogue Quality Cont.

Participants were asked to rate quests on these qualities:

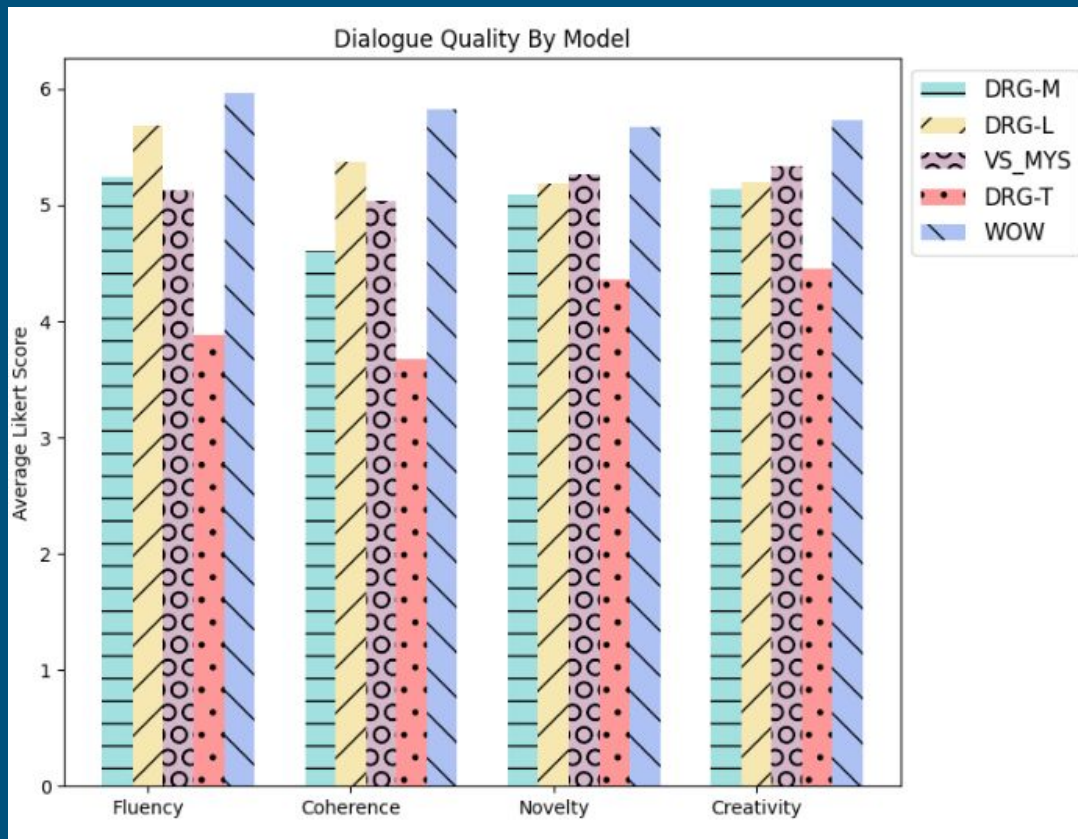
Fluency: The dialogue makes use of correct English

Coherence: The goal is clear from the dialogue

Novelty: The dialogue is written in a novel way

Creativity: The dialogue is creative

Study 1 : Dialogue Quality Cont.



Study 2: Dialogue Satisfaction

This Study was performed for finding out Players satisfaction level with the NPC Response.

Each Participant was shown three quests from 3 different sources after entering a text input prompt

Pipeline - The user input was sent into the pipeline method described earlier to create the quest tuple(Quest, Title, Dialogue)	WOW - A Random WOW quest is shown without regarding the input	4-Gram - Similar to an Large Language Model but create text by approximating. This was trained on WOW quest data.
---	---	---

Study 2: Dialogue Satisfaction Cont.

They then were asked asked 2 questions:

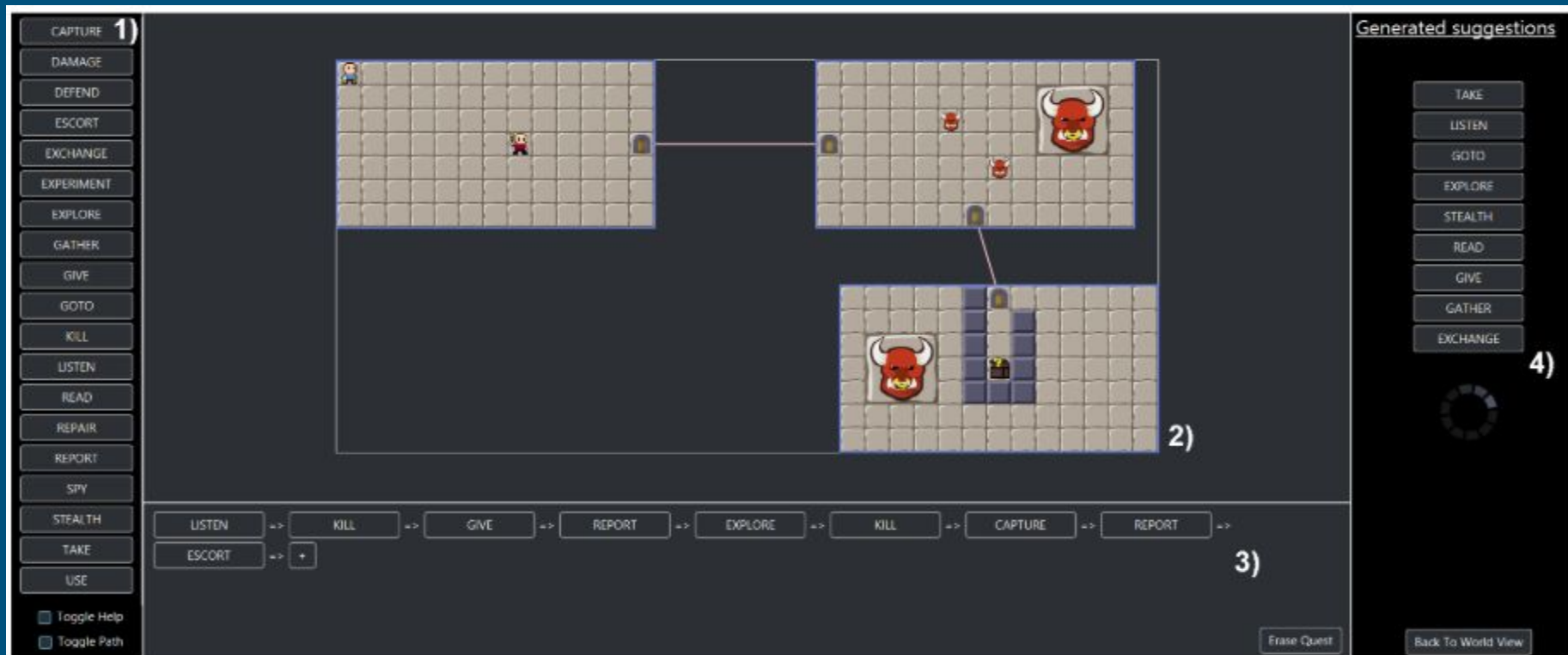
- Which prompt did you feel was most responsive to your input?
- Which prompt did you feel was most exciting/creative?

Model	Satisfaction	Input-Responsive	Satisfaction (when input contained 'dragon')	Input-Responsive (when input contained 'dragon')
WoW	197	159	14	5
Ours	125	169	14	21
N-Gram	58	52	0	2
Total	380	380	28	28

Questgram

- Questgram uses its base as a reference to Evolutionary Dungeon Designer (EDD)
- It is mixed initiative in the way that the AI system works together with the Developer
- Its purpose is to help developer makes quests and to help inspire creativity

EDD



(a) Overview of the GUI used for the design of quests in EDD. 1) The possible quest actions, 2) the dungeon created thus far by the designer, 3) the quest sequence, and 4) the suggestions from the grammar.

Quest Actions

Action	Prerequisites in [16]	Prerequisites in EDD
Capture	"Somebody is there"	A NPC or boss/enemy must be placed.
Damage	"Somebody or something is there"	An item or NPC must be placed.
Defend	"Somebody or something is there"	An item or NPC must be placed.
Escort	"Somebody is there"	A NPC must be placed.
Exchange	"Somebody is there, they and you have something"	A NPC and an item must be placed (requires two positions).
Experiment	"Something is there"	An item must be placed.
Explore	"none"	An available floor tile.
Gather	"Something is there."	An item must be placed.
Give	"Somebody is there, you have something."	A NPC and an item must be placed (requires two positions).
Goto	"You know where to go and how to get there."	An available floor tile.
Kill	"Somebody is there."	A boss/enemy must be placed.
Listen	"Somebody is there."	A NPC must be placed.
Read	"Somebody is there."	A NPC must be placed.
Repair	"Somebody is there."	A NPC must be placed.
Report	"Somebody is there."	A NPC must be placed.
Spy	"Somebody or something is there."	A NPC or boss/enemy must be placed.
Stealth	"Somebody is there."	A NPC or boss/enemy must be placed.
Take	"Somebody is there, they have something."	A NPC and an item must be placed (requires two positions).
Use	"There is something there."	An item must be placed.

Quest Actions Cont.

The screenshot displays a quest editor interface. On the left is a vertical list of actions: CAPTURE, DAMAGE, DEFEND, ESCORT, EXCHANGE, EXPERIMENT, EXPLORE, GATHER, GIVE, GOTO, KILL, LISTEN, READ, REPAIR, REPORT, SPY, STEALTH, TAKE, and USE. The 'EXPLORE' action is currently selected. The main area shows a grid-based map with two yellow icons connected by a line. At the bottom left, there is a red shield icon with a plus sign and the text: "Try adding different tiles to enable more actions". At the bottom right, there are two buttons: "Erase Quest" and "Back To World View". On the far right, a sidebar contains the text "Generated suggestions" and an "EXPLORE" button.

CAPTURE
DAMAGE
DEFEND
ESCORT
EXCHANGE
EXPERIMENT
EXPLORE
GATHER
GIVE
GOTO
KILL
LISTEN
READ
REPAIR
REPORT
SPY
STEALTH
TAKE
USE

Toggle Help
Toggle Path

Generated suggestions
EXPLORE

Erase Quest
Back To World View

Quest Grammar

2 general categories for the grammar production rules of quests:

1. Motivation: which is more of a quest/questline that has a more emotional reason to the quest

Production rules	Actions
knowledge*	["<get>","<go_to>","give"], ["<spy>"], ["<go_to>","listen","<go_to>","report"], ["<get>","<go_to>","use","<go_to>","give"]
comfort*	["<get>","<go_to>","give"], ["<go_to>","damage","<go_to>","report"]
reputation*	["<get>","<go_to>","give"], ["<go_to>","<kill>","<go_to>","report"], ["<go_to>","<go_to>","report"]

Quest Grammar Cont.

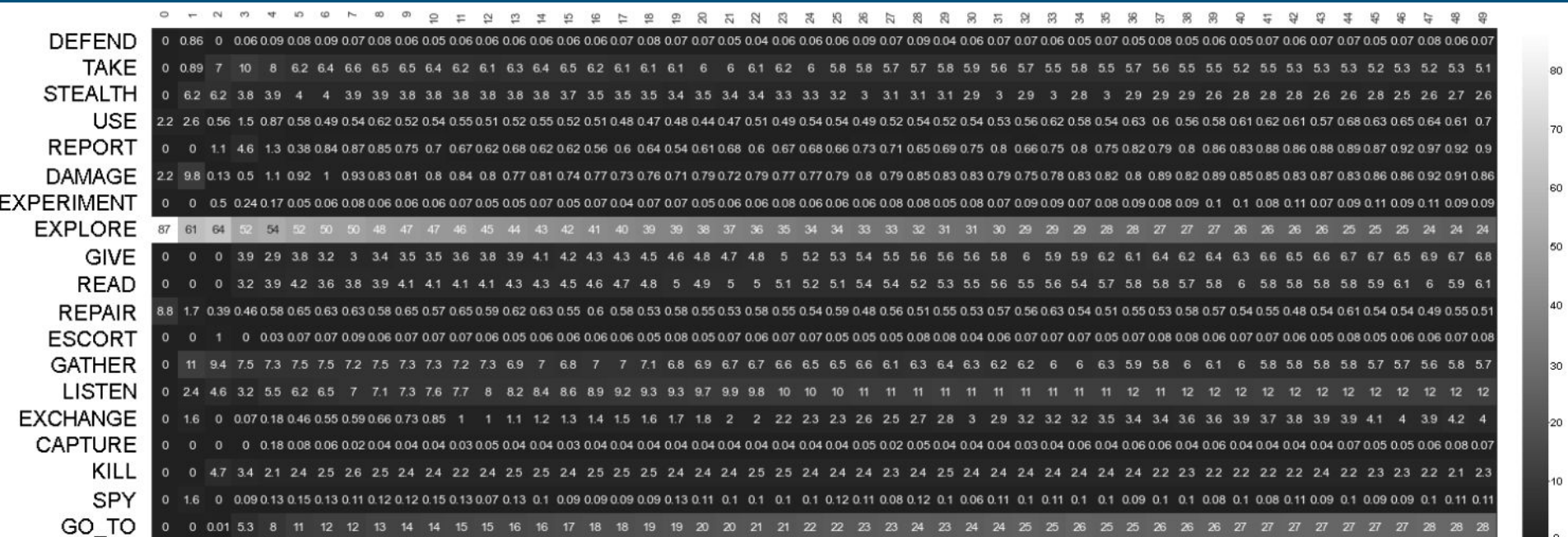
2. Non-Motivation: The more simple commands to progress the story such as “go_to” or “Explore”

```
go_to      ["explore"], ["<learn>","go_to"]
learn      ["<go_to>","<subquest>","listen"],
           ["<go_to>","<get>","read"],
           ["<get>","<subquest>","give","listen"]
get        ["<steal>"], ["<go_to>","gather"],
           ["<go_to>","<get>","<go_to>","<subquest>","exchange"]
```


Study 1: expressive Analysis

- ❑ They wanted to test the expressivity and diversity of the quest generated through the system
- ❑ They generated 100,000 quests with a maximum of 50 quest actions each.
- ❑ They then mapped out how often the quest actions occurred

Study 1: expressive Analysis Cont.



Study 2: User Study

6 Developers were asked to use the system and rate it using 3 different methods of quest: Manual, Automatic, Mixed-Initiative

Manual

- This was mostly done so that the Developers could get used to the tool and give feedback on it
- They describe the tool to be easy to use and still able to make the quests they want
- One issue was the inability to edit the questline

Study 2: User Study Cont.

Automatic

- Most Developers remarked saying that the tool felt illogical
- The tool would target random tiles and NPCs with no purpose
- They felt that the tool took away the freedom of creating the quest and overall complicated the process

Study 2: User Study Cont.

Mixed Initiative

- Overall was described as useful and helpful
- Most often was used when there was an inspiration blockage
- Was also described to help increase the speed of quest generation
- Some arguments were still that the suggestions lacked cohesiveness

Conclusion

Pipeline Method

- Flaws: Small Knowledge base size and language model training issues
- Pros: responsive and flaws can be fixed

Mixed Initiative Method

- Flaws: not flexible, takes away creative options on its own, and is simplistic
- Pros: helps the developer when they are stuck and helps inspire them with new options

Future expansion



Questions?
