

УДК 004.946:793.7

10.25209/2079-3316-2023-14-4-123-140



Генерация сюжетной составляющей для настольной ролевой игры

Диана Антоновна **Чельшева**^{1✉}, Аркадий Викторович **Ключиков**²

¹ СГТУ им. Ю.А. Гагарина, Саратов, Россия

² ФГБОУ ВО Вавиловский университет, Саратов, Россия

✉ diana.chelysheva.98@mail.ru

Аннотация. Проведён анализ предметной области и актуальности настольных ролевых игр, а также их влияние на сферы деятельности человека. Изучена механика игрового процесса настольной ролевой игры Подземелья и драконы. Рассмотрены программные решения для создания сюжетного описания локаций, в том числе графическое представление карт и одностраничные приключения. Реализованы дизайн и адаптивный пользовательский интерфейс. Сформирована архитектура базы данных, а также реализована в формате JSON-файлов. Разработана архитектура модели, а также произведена её настройка. Описан блочный способ построения сюжета. Разработан алгоритм работы приложения по генерации сюжетной составляющей. Сформированы списки сюжетных составляющих, а также реализована логика случайного выбора участков описания с проверкой на адекватность и сочетаемость описания.

Ключевые слова и фразы: генерация сюжета, настольная ролевая игра, «Подземелья и Драконы», мобильное приложение, пользовательский интерфейс, блочный способ построения сюжета, база данных

Для цитирования: Чельшева Д. А., Ключиков А. В. *Генерация сюжетной составляющей для настольной ролевой игры* // Программные системы: теория и приложения. 2023. Т. 14. № 4(59). С. 123–140. https://psta.psiras.ru/read/psta2023_4_123-140.pdf

Введение

Популярность настольных игр на сегодняшний день очень высока [1]. По данным игрового сайта Roll20, в разные настольные ролевые игры (НРИ) [2] играет 70 процентов пользователей. В НРИ участники устно описывают действия своих персонажей, основываясь на их особенностях. Самой распространённой среди таких игр является Dungeons Dragons (DND). По данным сайта Dungeon Vault, DND является популярной и востребованной, её сообщество насчитывает 13.7 млн. игроков по всему миру. В области НРИ ведутся разработки по графической интерпретации локаций и одностраничные приключения [3].

При написании игровых сценариев возникает ряд трудностей, замедляющих процесс повествования. Среди них — проблема выбора места, времени и условия действия, поскольку игра является вариативной в сюжетной составляющей, а порядок действий случаен. Эти аспекты усложняют задачу игрового мастера.

Рассмотрим сценарный подход с графической интерпретацией карт. Мастер игры описывает свой сценарий с использованием нарисованной карты, где изображены ключевые точки сюжетных событий.

Из видов карт выделяют разделенные на:

- этажи;
- комнаты (рисунок 1).

Одностраничные приключения создаются для личного использования мастером игры. Такое представление информации позволяет повысить вариативность сюжетных линий и визуализировать сюжет для быстрой ориентации в нем.

Чтобы упростить процесс генерации сюжетных аспектов, разработчики в сообществе любителей НРИ создали инструменты автоматической генерации. В настоящий момент существуют приложения и веб-ресурсы генерирующие сюжетные элементы для игр [4], чтобы сократить время на подготовку для игрового мастера.

Современные решения не способны генерировать описание локации совместно с функциями настройки сеттинга, местности, ключевых персонажей и предметов. В связи с этим предлагается разработать приложение для создания сюжетной составляющей игры. Среди доступной информации пользователю предоставляется описание: персонажей (героев), квестов и добычи, а также изображения описываемой территории, где текст генерируется блоками.

В липких лапах М'куса

Прохладная завязка:

На обычно безопасной дороге жарким солнечным утром 1кб гоблины из клана "Сопливые носы" пытаются украсть у героев медикаменты. Они явно чем-то болель.

Игра начинается в момент, когда персонажи игрово замечают воршиек. Гоблины будут сражаться, если возникнет угроза их жизни, и убегают после первого раунда боя, их можно уговорить или подкупить, чтобы узнать некоторые детали происходящего.

Проследуйте в город.

Городок Нус:

Шахтерский город, все жители которого увлечены местным фестивалем на рыночной площади. В течение дня почти все люди заблевают хворью, вызывающей чрезмерное выделение слизи. Герои делают спасбросок средней сложности против болезни каждый раз, когда они пьют воду.

Все эти массовые заболевания, похоже, никого не беспокоят.

Вечером все жители направляются в заброшенную шахту (храм), чтобы местные жрецы "излечили" их.

Заброшенная шахта – святилище носоподобного бога М'куса (Комната 1). Жрецы бесплатно раздают зелья, выводящие слизь из носов людей и "исцеляющие" их. Вся слизь стекает в желоб и в виде слизинов уплывает внутрь статуи большого носа, чтобы затем проникнуть дальше в святилище, где служители культа превращают всю собранную субстанцию в живую Слизь (Ooze), для продажи заинтересованным лицам.

Все это повторится ежедневно.

Если герои останутся в городе на другой день, то снова будет риск заболеть, т.к. больные люди-рыбы все еще загрязняют воду. Их можно остановить (см. описание подземелья – комната 8).

Скровища и вещи на продажу (1к10)

1-5: вещи или монеты стоимостью 1-5 золотых (ЗМ).

6-7: Втулка для чистки носа (50 ЗМ)

Всасывает слизь в пределах 1 фута (30 см), исцеляет больных на 1к6 ходов, через 3+1к6 дней из нее выходит малый Золотистый студень (Ochre jelly).

8: Щекокающая стрела (50 ЗМ)

Если стрела пролетит в пределах 15 футов (4,5 м) от существа, оно чихнет, выдав свое местоположение.

9: Исцеляющее полотенце (100 ЗМ)

Можно использовать во время короткого отдыха. Исцеляет любую хворь. После использования надо тщательно промыть.

10: Жезл кашля (150 ЗМ), 3 заряда

Один заряд заставляет каждое существо в пределах 60 ф (18 м) сильно кашлять (получая помеху и раскрывая свою позицию).



Подземелье: Шахты М'куса

Дверей нет. В зонах 1,2,3 и 4 есть фанелы, дающие слабый свет, области 7 и 8 покрыты свечами с мхом (тусклый свет).

1. Вход / Статуя Носа – 1к4 служителей культа, крошечное существо может пройти через ноздрю и попасть в центр комнаты 4.
2. Главный зал – пара ящиков с липкими мантиями или одним из сокровищ.
3. Жилые помещения – 2к4 служителей культа, спят или читают, один из них – маг.
4. Лаборатория слизи – 1к4 служителей культа (один маг), пытающихся превратить липкую субстанцию в Черную слизь (Black pudding). Через 1к4 хода у них это получится, если им не помешать.
5. Ямы для экспериментов – здесь магическим образом заперты 2 человека. Их можно использовать для замены погибших героев или же в виде монстров:
Хозяин слизи (слабый гуманоид, клинически безумен). 1 из 8 шанс/ход, извергнуть малый Золотистый студень (Ochre jelly). Ямы открываются, если маг убил или если магия рассеяна соответствующим заклинанием.
6. Лабиринт. Желатиновый куб движется всегда вдоль правой стены и не может использовать лестницы, занимает весь коридор по ширине, наверху есть зазор в 3 фута (1 метр) и торчат несколько потолочных балок.
7. Затопленная шахта – заброшена в спешке, драгоценные камни и минералы стоимостью 5к4*10 золотых можно добыть при помощи кирки; легкая проверка Интеллекта + бонус за наличие инструментов, при провале проверки произойдет обрушение (1к10 дробящего урона).
8. Пещера колонии – 2к6 зараженных людей-рыб (один из них – жрец) поклоняются М'кусу. Если победить их или опрокинуть статую в центре комнаты, то заражение прекратится, и слизь (и сам М'кус) исчезнет.
Жрец может призвать (сложная магическая проверка) *Аватара М'куса* – парящую в воздухе Черную слизь (Black pudding) в форме носа, атакующую выстрелами из ноздрей.

Рисунок 1. Пример одностраничного приключения

1. Разработка пользовательского интерфейса

1.1. Постановка задачи

В связи с выявленной проблематикой вариативности игры, практическая цель исследования заключается в сокращении затрачиваемого времени написания сюжетной составляющей настольной ролевой игры.

Для достижения цели поставлены следующие задачи:

- разработать дизайн пользовательского интерфейса программы;
- создать дизайн взаимодействия с пользователем (UX – User Experience) – создание диаграммы пользовательского пути (User flow);
- реализовать графическую часть интерфейса (UI – User Interface);
- разработать концепт интерфейса в онлайн-сервисе Figma;
- реализовать клиентскую часть приложения (frontend) на игровом движке Unity;
- разработать алгоритм блочной генерации сюжета.

Объектом исследования является процесс разработки интерфейса мобильного приложения для генерации сюжетной составляющей настольной ролевой игры. Предмет исследования – программа для генерации сюжетной составляющей настольной ролевой игры.

1.2. Описание решения

Перед реализацией визуальной части приложения требуется разработать дизайн взаимодействия с пользователем, где UX – это впечатление пользователя от работы с интерфейсом приложения. Для UX важны: дизайн, архитектура приложения, отзывчивость интерфейса и сложности формулировок и терминологии.

Основной задачей на этом этапе являлось создание диаграммы пути пользователя. Flow-диаграммы, отображающие полный путь, движения пользователя при использовании продукта.

После создания UX разрабатывается графическая часть интерфейса.

Графическая часть интерфейса работает в связке с UX-основой приложения, удовлетворяя при этом эстетические желания пользователя. Рассмотрим ключевые этапы создания UI.

На этапе разработки реализован стилистический аспект приложения. Принято решение сфокусироваться на ассоциации игроков DND со старыми компьютерными RPG. Это отражено в выборе шрифтов, цветов и стиля пиксель-арт, используемых в приложении.

Создано несколько коллажей с текстурами, элементами типографики, цветовой палитрой и цитатами (мудборд).

Для реализации текстовых блоков использовался шрифт Fifaks 1.0 dev1.

После разработки дизайна UX/UI реализована клиентская часть с использованием игрового движка Unity. На этом этапе выполнена верстка пользовательского интерфейса (рисунок 2).

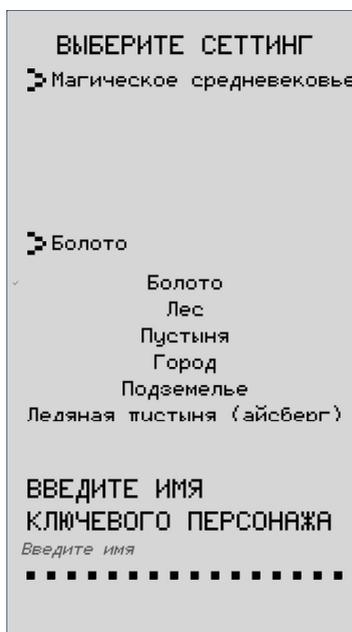


Рисунок 2. Выбор сеттинга

Unity предоставляет инструменты для настройки пользовательского интерфейса и возможность отображения 2D графики с использованием компонента Sprite Renderer. Для адаптивной верстки интерфейса приложения в движке можно установить опорное разрешение, выбрать ориентацию экрана и указать пропорции для изменения ширины и/или

высоты дочерних элементов (скалирование — «scaling», масштабирование). Применяемые инструменты и процесс верстки показаны на рисунке 3.

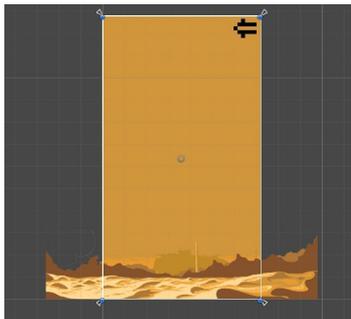


Рисунок 3. Верстка интерфейса в игровом движке Unity
Интерфейс адаптивный (рисунок 4). Для изменения размеров элемен-

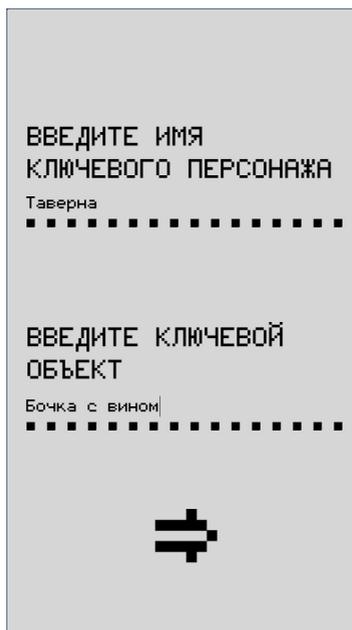


Рисунок 4. Поля ввода входных параметров

тов, а также расположения элементов в зависимости от разрешения экрана устройства используются якорные точки и инструменты скалирования.

Результат работы визуализируется во вкладке Game, где возможно

варьировать разрешение тестируемой сцены или устройства. Финальный вариант интерфейса приложения с учетом дизайна и верстки отображен на рисунке 5.

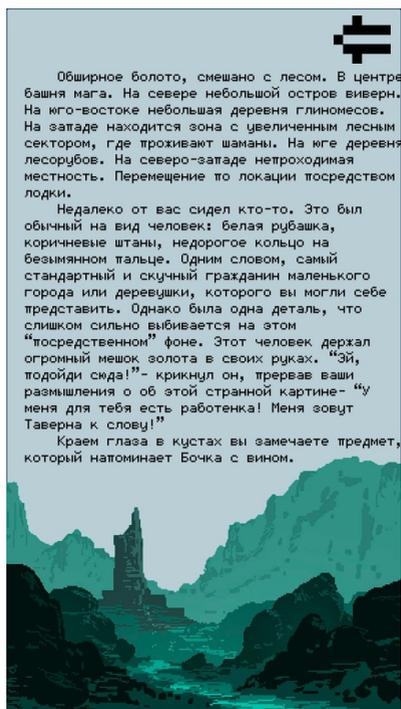


Рисунок 5. Финальный вариант интерфейса

Интерфейс создан с использованием программного обеспечения Photoshop, Figma и Unity.

Далее разработан и описан алгоритм работы приложения по генерации сюжетной составляющей.

В процессе исследования разработан алгоритм [5] работы приложения по генерации сюжетной составляющей [6–8]. На стартовом экране пользователю необходимо ввести входные характеристики для создания истории (генерации сюжета).

Реализован функционал приложения. Переключение между экранами, выбор сеттинга, выбор локации, ввод информации о предмете и персонаже.

Сформированы списки сюжетных составляющих, а также реализована логика случайного выбора участков описания с проверкой на адекватность

и сочетаемость описания.

2. Разработка программной составляющей приложения

Unity — игровой движок, в котором скрипты функционирования игровых объектов интерпретируются на языках высокого уровня C# и JavaScript (JS). Однако для обработки кода на JS требуется дополнительный интерпретатор, поэтому основным языком проекта выбран C#.

Архитектура Unity основана на компонентно-ориентированной системе, в которой игровые объекты объединены в сущности с различными компонентами. C# является объектно-ориентированным и жестко типизированным языком.

2.1. Анализ существующих технических решений

Выбор метода зависит от конкретных задач и требований генерации историй и сюжетов. В таблице 1 приведён сравнительный анализ под-

ТАБЛИЦА 1. Анализ существующих технических решений

№	Способ генерации сюжета	Описание метода	Языковая модель и архитектура	Набор данных (Dataset)
1	Event2event	Разбиение набора событий на последовательности и генерация этих событий с использованием языковой модели	Рекуррентные нейронные сети (RNN)	Общедоступный
2	event2sentence	Преобразование структурированных данных в естественные предложения	RNN	Общедоступный
3	УРНГ	Управляемая и редактируемая нейронная генерация сюжета	Control-and-Edit Transformer (CET)	Сюжетное описание фильмов из Википедии
4	Desc2Story	Автоматическое создание и развитие истории, основываясь на предоставленной информации или инструкциях	RNN	Общедоступный набор данных VIST (Visual Storytelling Dataset)
5	Блочный способ	Блок — часть сюжета	transformer — GPT2 от OpenAI	Возможна тренировка на WritingPrompts dataset, или на авторской БД

ходов [9, 10]: метода event2event, event2sentence, УРНГ, Desc2Story и предложенного способа блочной генерации сюжета.

- (1) Event2event. Метод основан на генерации событий и их последовательностей. Модель способна прогнозировать возможные последствия каждого события в контексте истории.

Преимущества: формирует последовательности событий для автоматической генерации историй и сюжета с логической структурой и связностью между ними, моделирует и предсказывает следующее событие на основе предыдущих.

Ограничения: метод менее информативный в сравнении с Event2sentence, т.к. генерирует только последовательность без подробных описаний. Модель не всегда гарантирует логичный и связный сюжет для всех входных данных.

- (2) Event2sentence. Предложения, описывающие отдельные события, генерируются с использованием модели, обрабатывающей входные последовательности событий и создающей связные и правильные предложения, соответствующие контексту.

Преимущества: создание более информативных и содержательных описаний событий в сравнении с Event2event, превосходящих простую последовательность, например, при необходимости генерации детальных описаний. Генерирует истории с сохранением сюжетной структуры.

Ограничения: для успешной работы требуется модель, способная понимать семантику и взаимосвязи между событиями. Уделяет внимание логической структуре и связности меньше, чем блочный способ. Имеет ограничения по переводу типов событий в предложение. Требуются большие объемы подходящих данных (больше, чем у блочного способа или метода event2event).

- (3) УРНГ – управляемая и редактируемая нейронная генерация сюжета.

Преимущества: позволяет вмешиваться в процесс генерации текста, искать подходящие части текста и редактировать их по своему усмотрению. Контролируемая генерация текстовых описаний.

Недостатки: ограниченность вариативности генерации приводит к повторению или предсказуемости. Сложность обучения и моделирования связей между историями. Имеет ограниченную способность генерировать сложные истории с нелинейной структурой.

- (4) Desc2Story. Построен на генерации сюжетов на основе описания или аннотации. Модель получает входные данные в виде описания сюжета или набора ключевых слов, и генерирует историю или сюжет.

Преимущества: генерация истории из коротких описаний создаёт разные сценарии и сюжетные линии. Короткие описания в исходной последовательности могут быть независимыми, что обеспечивает гибкость в создании истории.

Ограничения: недостаток информации в наборе данных. Ограничения контекста могут привести к поверхностному представлению сюжета. Сложность в создании закономерностей.

- (5) Блочный способ построения сюжета. Предложенный подход основан на методе генерации сюжетов с использованием блоков, где история разделяется на отдельные блоки или сцены, а затем генерируется содержимое каждого блока в рамках происходящих в нем событий. Каждый блок представляет отдельное действие, сцену или последовательность событий, и генерация сюжета осуществляется блок за блоком. Блочный подход позволяет определить ключевые моменты и переходы между элементами, что создаёт более качественные и интересные истории.

Преимущества: создаёт структурированные сюжеты, разбитые на блоки или сцены. Уделяет внимание разделению и категоризации информации по этапам развития сюжета. Обеспечивает контролируемый процесс генерации, т.к. основана на предварительно созданных блоках. Такой подход применяется при необходимости управления развитием сюжета.

Ограничения: требует предварительного разбиения истории на блоки или сцены. Ограничение гибкости и творчества, т.к. имеет строго описанную структуру. Создание авторской базы данных потребует поиска, обработку и структурирования информации, провоцируя временные и материальные издержки.

В результате проведенного анализа принято решение выбрать блочный способ построения сюжета. Одним из главных преимуществ блочного подхода является его структурированность. Каждый блок имеет определенные функции и связи с другими, что обеспечивает логическую последовательность и качественное развитие сюжета.

2.2. Разработка алгоритма функционирования приложения

Началом работы в приложении является стартовый экран, на котором пользователю предлагается указать характеристики для создания сюжета. Пользователь может выбрать сеттинг и локацию из списка предложенных значений. Затем требуется ввести имя ключевого персонажа и название объекта.

В соответствии с выбранным сеттингом и локацией определяется стиль текстового описания. В зависимости от выбранной локации, приложение подбирает описание местности из предварительно разработанного списка элементов и фоновое изображение для следующего экрана. Затем, пользователь может дополнить описание, вводя информацию о персонаже и объекте, при этом значения также зависят от выбранного сеттинга и локации.

Результатом является текстовое описание, которое появляется на экране №2, и которое может смещаться вверх или вниз в зависимости от его длины.

2.3. Проектирование базы данных на основе анализа предметной области

Информация об одностраничных приключениях хранится в JSON-файлах и применяется в дальнейшем для создания базы данных (БД). В файлах содержится следующая информация: NPC, Actions, Locations, Quests и Items (рисунок 6).

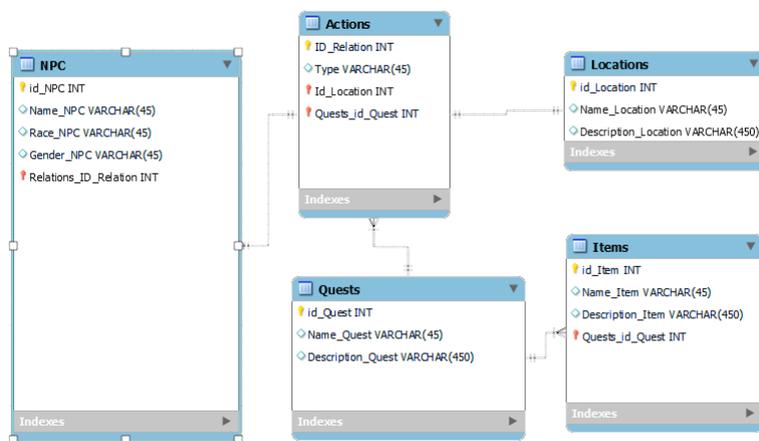


Рисунок 6. ER-диаграмма

БД состоит из различных сюжетных элементов, оцифрованных и сохраненных в формате JSON. Каждый JSON-файл содержит описание и список ключевых моментов, характеризующих события и персонажей в составе сюжета, отделяя события от менее важных, чтобы обеспечить более качественный результат. БД является авторской, а ее архитектура основана на одностраничных приключениях. Это позволяет обеспечить более легкий и удобный доступ к сюжетным элементам и ключевым моментам, улучшая понимание структуры и логики повествования. Формат JSON также обеспечивает простоту и удобство работы с БД для дальнейшего использования, хранения и обработки ее содержимого.

2.4. Описание используемого технического решения

Блочный способ построения сюжета состоит из четырех полей: персонажи, действия, описание локации и квесты. Каждое из этих полей

содержит текстовое описание, комбинирующееся вместе, чтобы создать единую историю.

Персонажи – главные герои или участники истории, имеющие свои уникальные характеристики, определяющие роль в сюжете.

Действия – последовательность событий, происходящих в истории и развивающих сюжет, физические действия персонажей, их мысли и эмоции.

Описание локации – текстовое описание места, где происходят события истории. Состоит из информации о физической обстановке, атмосфере, архитектуре локации. Создаёт образное представление о месте действия.

Квесты – цели и задачи, поставленные перед персонажами. Развивают сюжет, достигая определенных результатов.

Блочный способ создаёт связные истории, основываясь на заранее определенных блоках информации и предварительно обученной модели. Fine-tuning – точная настройка, подход к переносу обучения в deep learning, при котором веса предварительно обученной модели обучаются на новых данных¹. После определения архитектуры (рисунок 7) модель обучалась

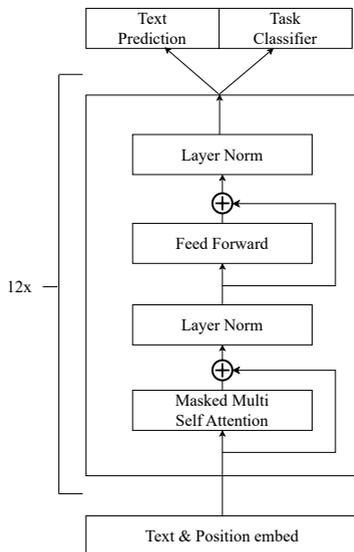


Рисунок 7. Архитектура модели

¹См. <https://www.datacamp.com/tutorial/fine-tuning-gpt-3-using-the-open-ai-api-and-python>

на тренировочном наборе данных, затем оценивалась на тестовом, где для достижения результата потребовалась дополнительная настройка модели.

Использование fine-tuning модели позволяет снизить затраты на вычисления и использовать современные языковые модели без необходимости обучения их с нуля. Transformers предоставляют доступ к моделям для различных задач.

При настройке модели указываются параметры:

- (1) `Output_dir`, определяет директорию, в которой хранятся результаты тренировки модели. В этой директории сохранены: словарь используемого токенизатора, конфигурация модели, контрольные точки, значения метрик и состояние обучения.
- (2) `Num_train_epochs`, устанавливает общее количество эпох для обучения модели. Чем больше эпох, тем больше времени потребуется для тренировки модели.
- (3) `Per_device_train_batch_size`, определяет размер пакета данных, используемый для обучения модели. Этот параметр влияет на скорость обучения и использование памяти.
- (4) `Per_device_eval_batch_size`, устанавливает размер пакетов данных, используемый для оценки модели, что влияет на скорость и использование памяти.
- (5) `Warmup_steps` – количество шагов разминки для установления темпов обучения. Используется низкая скорость обучения в начале тренировки модели, а затем скорость постепенно увеличивается.
- (6) `Weight_decay`, определяет силу снижения веса в сети и влияет на регуляризацию модели. Более высокое значение этого параметра приводит к большей регуляризации.
- (7) `Logging_dir` – директория, хранящая логи тренировки модели. Логи содержат информацию о потерях, точности и других метриках, а также обновляются через определенное количество шагов.
- (8) `Logging_steps`, устанавливает количество шагов обновления между записью логов, что позволяет контролировать частоту обновления логов и сохранять информацию о прогрессе тренировки.

Произведена настройка модели. Fine-tune сокращает ресурсы, необходимые для обучения новой модели с нуля, т.к. он использует уже существующую модель в качестве отправной точки. Время генерации составило 12 секунд.

2.5. Реализация функционала мобильного приложения

Start и Update являются основными методами, используемыми в игровом движке Unity. Start отвечает за создание больших списков, состоящих из описаний местоположений и настроек. Update проверяет недопустимые символы в полях персонажа и объекта и вычисляет длину выражений, используя методы CheckCharacters и CheckObjects (рисунок 8).

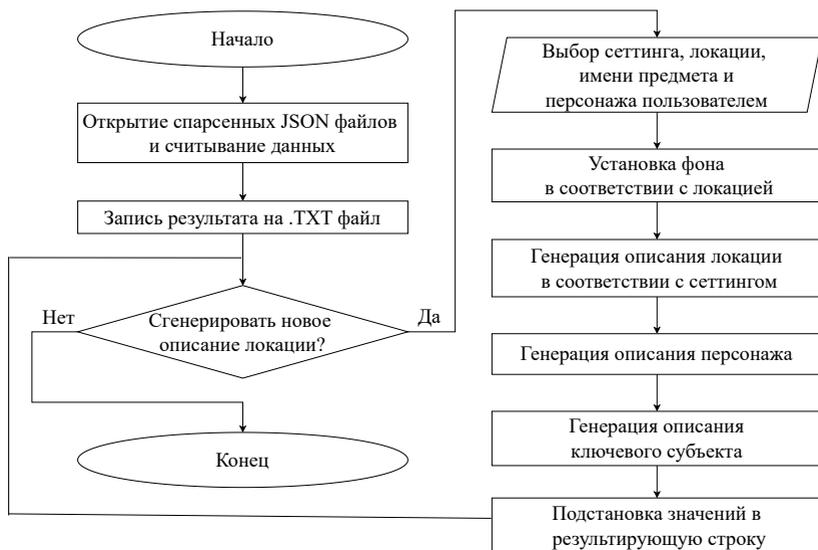


Рисунок 8. Блок-схема решения

Метод SummarySetActive отображает описание на странице и переключает на второй экран. Он привязан к кнопке в пользовательском интерфейсе и срабатывает при нажатии. Кроме того, SummarySetActive выбирает значение для отображения на экране на основе случайно сгенерированного ввода.

При генерации нового сюжета, выбирается случайный набор элементов из всех доступных JSON файлов, чтобы создать цельный и логичный сюжет. Для извлечения фрагментов сюжетов из различных JSON файлов используется парсинг, комбинирующий их в новые уникальные истории.

Заключение

В ходе исследования проведена работа по созданию сюжетно-генеративного приложения для настольной ролевой игры Dungeons&Dragons, а именно разработаны:

- UX/UI приложения;
- БД одностраничных приключений;
- алгоритм блочной генерации сюжета.

Разработанное приложение способно генерировать сюжет на основе выбранных пользователем сеттинга и локации за 12 секунд.

Для подтверждения полученных результатов приложение протестировано Саратовским сообществом игровых мастеров DND.

Согласно результатам опроса фокус группы, большинство отзывов людей, принимавших участие в тестировании, были положительными. Время, необходимое для подготовки игры, сократилось в среднем с одной недели до нескольких часов.

Список литературы

- [1] Писаревская Д. Б. *Ролевые игры: пример «социализации» субъекта* // Этнографическое обозрение.– 2008.– № 1.– С. 8–18. *  [↑124](#)
- [2] Седых И. А. *Индустрия компьютерных игр-2020*.– Высшая школа экономики. Центр развития.– 2020.– 74 с.  [↑124](#)
- [3] Малыхина А. С. *Понятие о сюжетно-ролевой игре. Структурные элементы игры* // Молодой ученый.– 2020.– № 22(312).– С. 539–541. *  [↑124](#)
- [4] Сахибгареева Г. Ф., Кугуракова В. В. *Редактор интерактивной структуры для инструмента генерации сценарных прототипов* // Электронные библиотеки.– 2022.– Т. 24.– № 6.– С. 1184–1202.   [↑124](#)
- [5] Каюмов Б. И. *Проблемы визуализации разветвленных сюжетов компьютерных игр*.– Казанский (Приволжский) федеральный университет.– 2021.– 79 с. [↑129](#)
- [6] Сахибгареева Г. Ф., Бедрин О. А., Кугуракова В. В. *Раскадровка как одно из представлений сценарного прототипа компьютерных игр* // Электронные библиотеки.– 2021.– Т. 24.– № 2.– С. 408–444.   [↑129](#)
- [7] Сахибгареева Г. Ф., Кугуракова В. В. *Концепт инструмента автоматического создания сценарного прототипа компьютерной игры* // Электронные библиотеки.– 2018.– Т. 21.– № 3–4.– С. 235–249.  [↑129](#)
- [8] Сахибгареева Г. Ф. *Применимость разветвленных структур для генерации сценарных прототипов видеоигр* // 65-я Международная научная конференция Астраханского государственного технического университета (Астрахань, 26–30 апреля 2021 г.), Астрахань: Изд-во АГТУ.– 2021.– С. 596–600. *  [↑129](#)

- [9] Martin L., Ammanabrolu P., Wang X., Hancock W., Singh S., Harrison B., Riedl M. *Event representations for automated story generation with deep neural nets*, Thirty-Second AAAI Conference on Artificial Intelligence (February 2–7, 2018, New Orleans, Louisiana, USA) // Proceedings of the AAAI Conference on Artificial Intelligence.– 2018.– Vol. **32**.– No. 1.   ↑130
- [10] Das A., Verma R. M. *Can machines tell stories? A comparative study of deep neural language models and metrics* // IEEE Access.– 2020.– Vol. **8**.– Pp. 181258–181292.  ↑130

Поступила в редакцию 12.05.2023;
 одобрена после рецензирования 06.11.2023;
 принята к публикации 02.12.2023;
 опубликована онлайн 17.12.2023.

Рекомендовал к публикации

д.ф.-м.н. А. М. Елизаров

Информация об авторах:

Диана Антоновна Челышева

Студент, является автором трёх публикаций, участником четырёх международных молодёжных научно-практических конференций, победитель регионального конкурса "Студенческий стартап" (III очередь), работает над созданием проектов в Студенческом конструкторском бюро "Робототехника и интеллектуальные системы", сфера научных интересов - искусственный интеллект, дистанционное управление 3D печатью, программирование, робототехника, компьютерная графика.

Foto by D. A. Chelysheva

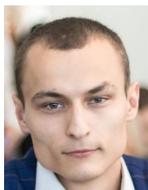


 0009-0002-4941-0767

e-mail: diana.chelysheva.98@mail.ru

Аркадий Викторович Ключиков

К.т.н., зав. кафедрой "Цифровое управление процессами в АПК" ФГБОУ ВО Вавиловский университет, директор Студенческого конструкторского бюро "Робототехника и интеллектуальные системы" СГТУ имени Гагарина Ю.А., сфера научных интересов - 3D дисплеи, робототехника, имитационное моделирование, программирование.



 0000-0002-8657-6486

e-mail: krok9407@mail.ru

Вклад авторов: *Д. А. Челышева* – 50.00% (идея, методология, создание стилевого файла, сбор материала); *А. В. Ключиков* – 50.00% (написание черновой версии, доработка и редактирование, визуализация, администрирование).

Авторы заявляют об отсутствии конфликта интересов.



Generation of a plot component for a tabletop role-playing game

Diana Antonovna **Chelysheva**¹, Arkady Viktorovich **Klyuchikov**²

¹ Saratov State Technical University, Saratov, Russia

² Vavilov University, Saratov, Russia

[✉] diana.chelysheva.98@mail.ru

Abstract. The analysis of the subject area and relevance of tabletop role-playing games, as well as their impact on the spheres of human activity, is carried out. The mechanics of the gameplay of the tabletop role-playing game Dungeons and Dragons have been studied. Software solutions for creating a plot description of locations, including a graphical representation of maps and one-page adventures, are considered. Implemented design and adaptive user interface. The database architecture has been formed, as well as implemented in JSON file format. The architecture of the model has been developed, as well as its configuration. The block method of plot construction is described. The algorithm of the application for generating the plot component has been developed. Lists of plot components are formed, as well as the logic of random selection of sections of the description with a check on the adequacy and compatibility of the description is implemented. (*In Russian*).

Key words and phrases: plot generation, tabletop role-playing game, "Dungeons and Dragons", mobile application, user interface, block method of plot construction, database

2020 *Mathematics Subject Classification:* 97P30; 68Q87

For citation: Diana A. Chelysheva, Arkady V. Klyuchikov. *Generation of a plot component for a tabletop role-playing game*. Program Systems: Theory and Applications, 2023, **14**:4(59), pp. 123–140. (*In Russ.*).

https://psta.psir.ru/read/psta2023_4_123-140.pdf

References

- [1] D. B. Pisarevskaya. “Role-playing games: a case of the “subculture” socialization”, *Etnograficheskoe obozrenie*, 2008, no. 1, pp. 8–18 (in Russian).
- [2] I. A. Sedyx. *Computer Games Industry-2020*, Vysshaya shkola ekonomiki. Centr razvitiya, 2020 (in Russian), 74 pp. 
- [3] A. S. Malyxina. “The concept of a plot-role-playing game. Structural elements of the game”, *Molodoj uchenyj*, 2020, no. 22(312), pp. 539–541 (in Russian). 
- [4] Saxibgareeva G. F., Kugurakova V. V.. “Interactive structure editor for scenario prototyping tool”, *Elektronnye biblioteki*, **24:6** (2022), pp. 1184–1202 (in Russian).  
- [5] Kayumov B. I.. *Problems of visualization of branched plots of computer games*, Kazanskiy (Privolzhskij) federal’nyj universitet, 2021 (in Russian), 79 pp.
- [6] G. F. Saxibgareeva, O. A. Bedrin, V. V. Kugurakova. “Storyboard as one of the representations of the scenario prototype of computer games”, *Elektronnye biblioteki*, **24:2** (2021), pp. 408–444 (in Russian).  
- [7] G. F. Saxibgareeva, V. V. Kugurakova. “The concept of automatic creation tool for computer game scenario prototype”, *Elektronnye biblioteki*, **21:3–4** (2018), pp. 235–249 (in Russian). 
- [8] G. F. Saxibgareeva. “Applicability of branched structures for generating scenario prototypes of video games”, *65-ya Mezhdunarodnaya nauchnaya konferenciya Astraxanskogo gosudarstvennogo texnicheskogo universiteta* (Astraxan’, 26–30 aprelya 2021 g.), Izd-vo AGTU, Astraxan’, 2021, pp. 596–600 (in Russian). 
- [9] L. Martin, P. Ammanabrolu, X. Wang, W. Hancock, S. Singh, B. Harrison, M. Riedl. “Event representations for automated story generation with deep neural nets”, Thirty-Second AAAI Conference on Artificial Intelligence (February 2–7, 2018, New Orleans, Louisiana, USA), *Proceedings of the AAAI Conference on Artificial Intelligence*, **32:1** (2018).  
- [10] A. Das, R. M. Verma. “Can machines tell stories? A comparative study of deep neural language models and metrics”, *IEEE Access*, **8** (2020), pp. 181258–181292. 