Архитектура Нейронной сети XXX

Функциональность

Давайте для начала пару слов о нейронной сети, которую команда “Mutex” проектировали и разрабатывали в рамках Лабораторной работы. Создавая нейронную сеть мы постарались создать что-то, что поможет контент криертерам вдохновалятся и создавать что-то новое, а также помогать поэтам, писателям вдохновляться. Именно поэтому мы сосредочились больше на обучении и генерации забавных текстов, ежели чем на интерфейсе.

Одним из главных преимуществ нашей нейронной сети является общедоступность и бесплатность. А также управление на интуитивном уровне, мы постарались все предусмотреть.

* 1 кнопка - сгенерировать текст
* Поле - поле вывода сгенерированного текста

Планируем добавить:

* Выбор темы - генерация текста по темам, которые будут предложены в приложение

Решение о старте разработки мы принимали преследуя следующие цели:

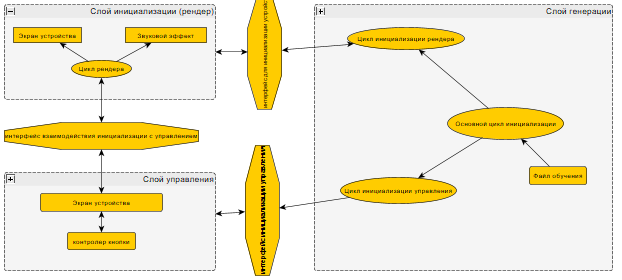
* В первую очередь нам была интересна концепция работы нейронных сетей
* Также мы преследовали цель изучения и развития наших способностей в программировании

Архитектура

Этап 1. Концепция

Первый, концептуальный, вариант архитектуры создавался нами после выбора языка программирования и платформы разработки нейронной сети. Он приведён на картине ниже, в этом варианте мы выделили слои будущего приложение:

* Слой инициализации
* Слой управления
* Слой генерации



При разработке нейронной сети первым был разработан слой генерации, так как при изучении вариантов и выборе темы мы пришли к решению, что слой генерации самый сложный и объёмный. Слой инициализации разрабатывался в последнюю очередь, так как было принято решение, что данный слой является менее объёмным и сложным, но изюминкой приложения при открытии будет звуковой эффект, который будет сопровождать пользователя при открытии. Слой управления был выделен отдельно с целью подчёркивания её простоты и понятности пользователям.

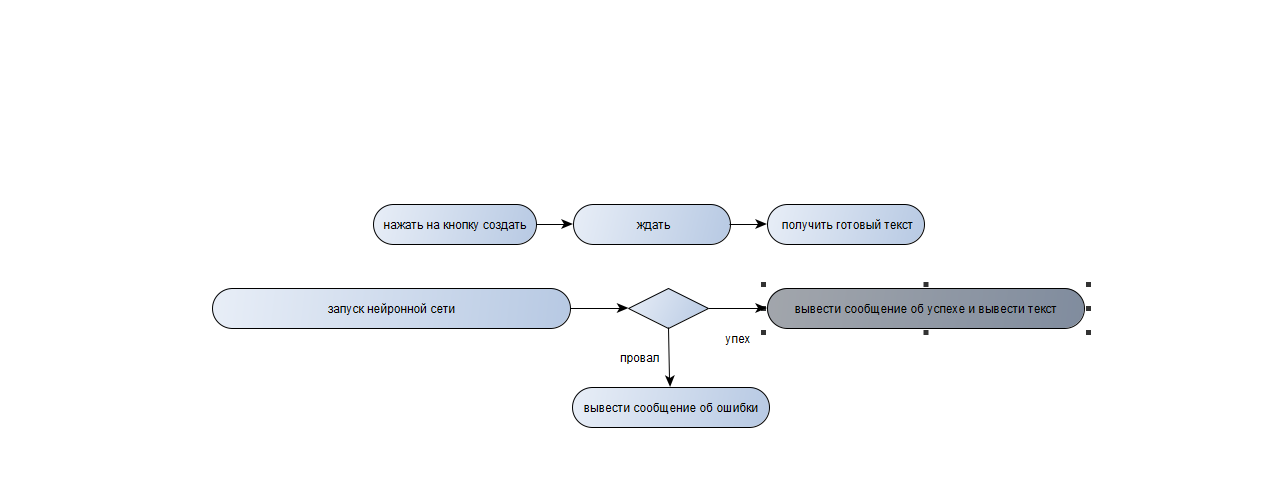
Хорошие идеи:

* Создание тематической генерации текста и возможность выбора тематики
* Улучшить слой генерации, после обучения нейронной сети, удаление файла обучения, в целях быстродействия нейронной сети (она будет уже обучена)

Промахи:

* Генерация хорошего и понятного текста не гарантировано

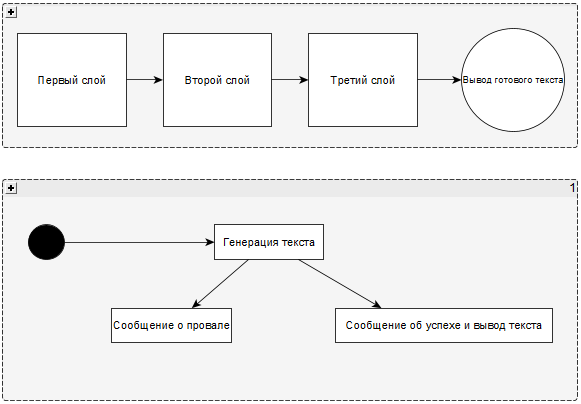
Этап 2. Проект

В процессе выбора языка программирования, был создан дизайн-документ и на его основе родился второй проект архитектуры. 

На данной диаграмме представлены все объекты логики и логические связи между ними. Данная диаграмма является общим изображением логики нейронной сети.

Общая архитектура — носит общий концептуальный характер, являясь, по сути, продолжением диаграммы этапа 1. На ней отображены:

* Компоненты системы (прямоугольники)
* Потоки данных (стрелки)
* Результат работы программы (кружок)



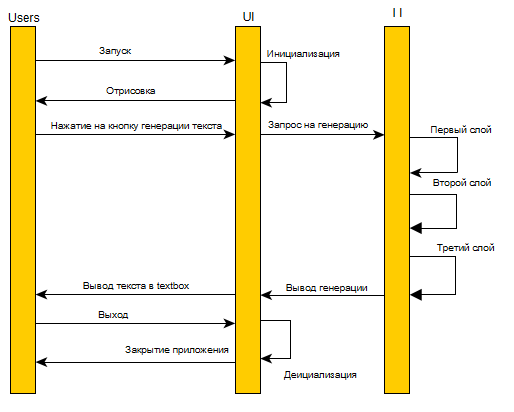
В целом концепция изменилась не сильно: также есть отдельный компонент для инициализации, далее выделены две основные подсистемы. После запуска приложения первым начинает работать компонент инициализации: происходит нажатие на кнопку и генерация текста, а также действия в зависимости от генерации текста.

Этап 3. Фактический результат

После разработки архитектуры есть вероятность, что

Этап 4. Запуск приложения

Диаграмма ниже отражает процедуру старта приложения (в той её части, которая контролируется разработчиком)



Загрузка приложения происходит при открытии приложения, также после открытия приложения пользователь нажимает на кнопку «Сгенерировать» после чего происходит генерация текста на основе обученной на книгах нейронной сети и выводит сгенерированный текст.

Этап 5. Запуск нейронной сети