Архитектура нейронной сети XX

Команда разработки “Mutex”

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:

* Выбор вариантов книги для генерации текста (Гарри Поттер, Шерлок Холмс).
* Кнопка «Ру» - автоматического перевода сгенерированного текста .
* Кнопка «Скопировать» - автоматического копирования сгенерированного текста в буфер.
* Кнопка «Удалить» - удаление, очищение поля вывода.
* Поле вывода - поле вывода сгенерированного текста.
* Поле выбора - выбор количества, ограничение длинны символов для генерации текста.
* Кнопка «Сгенерировать текст» - генерации текста.
* Кнопка пересгененрировать - пересгенерировка текста

Планируем добавить:

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

Решение о старте разработки мы принимали преследуя следующие цели:

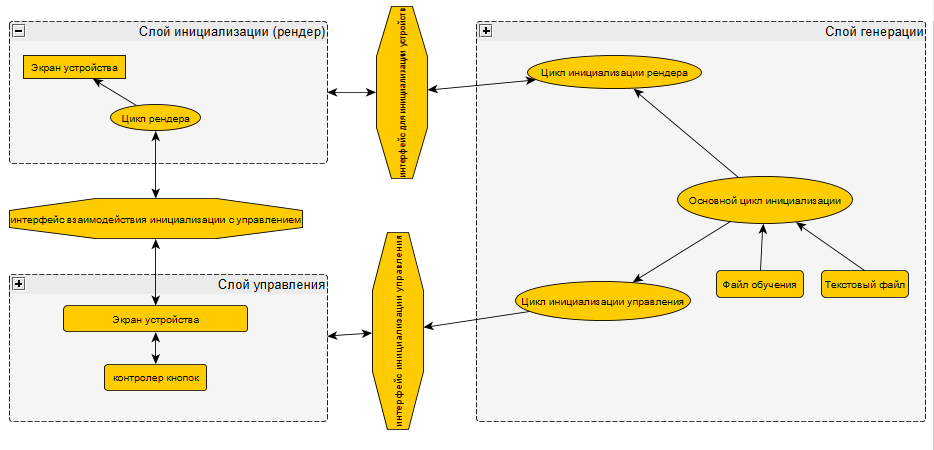
* В первую очередь нам была интересна концепция работы нейронных сетей
* Также мы преследовали цель изучения и развития наших способностей в программировании

Архитектура

Этап 1. Концепция

Первый, концептуальный, вариант архитектуры создавался нами после выбора языка программирования и платформы разработки нейронной сети. Но в процессе разработки были внесены изменения в нейронную сеть и она представлена на картинке ниже:

* 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. Слой управления был выделен отдельно с целью подчёркивания её простоты и понятности пользователям.

Хорошие идеи:

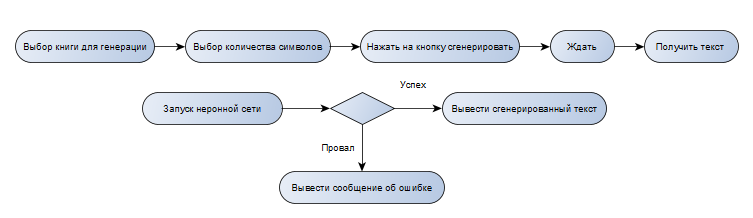
* Создание тематической генерации текста и возможность выбора тематики

Промахи:

* Генерация хорошего и понятного текста не гарантировано
* Плохой перевод с английского на русский

Этап 2. Проект

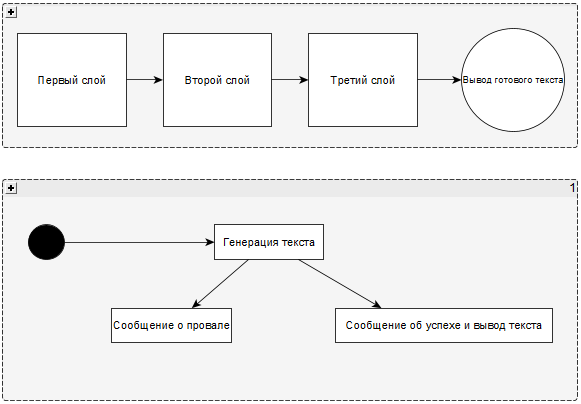
В процессе выбора языка программирования, был создан дизайн-документ и на его основе родился второй проект архитектуры.



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)

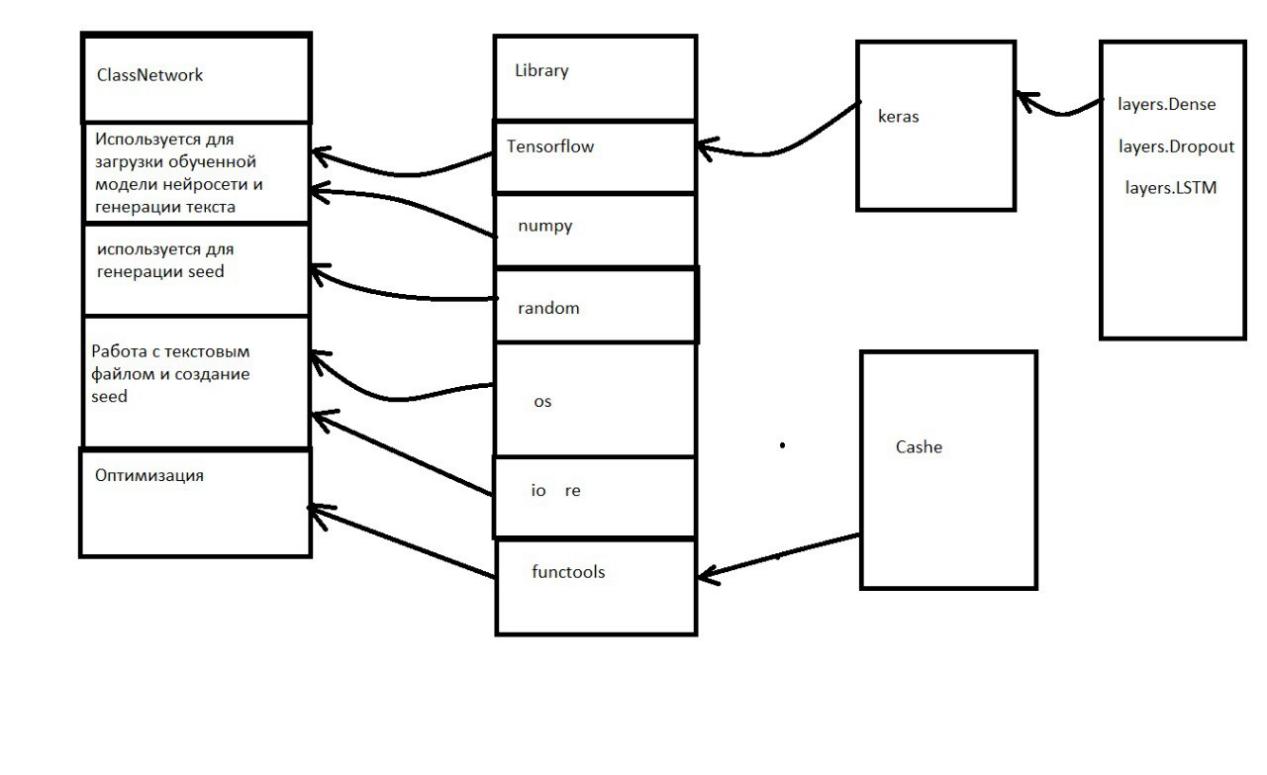


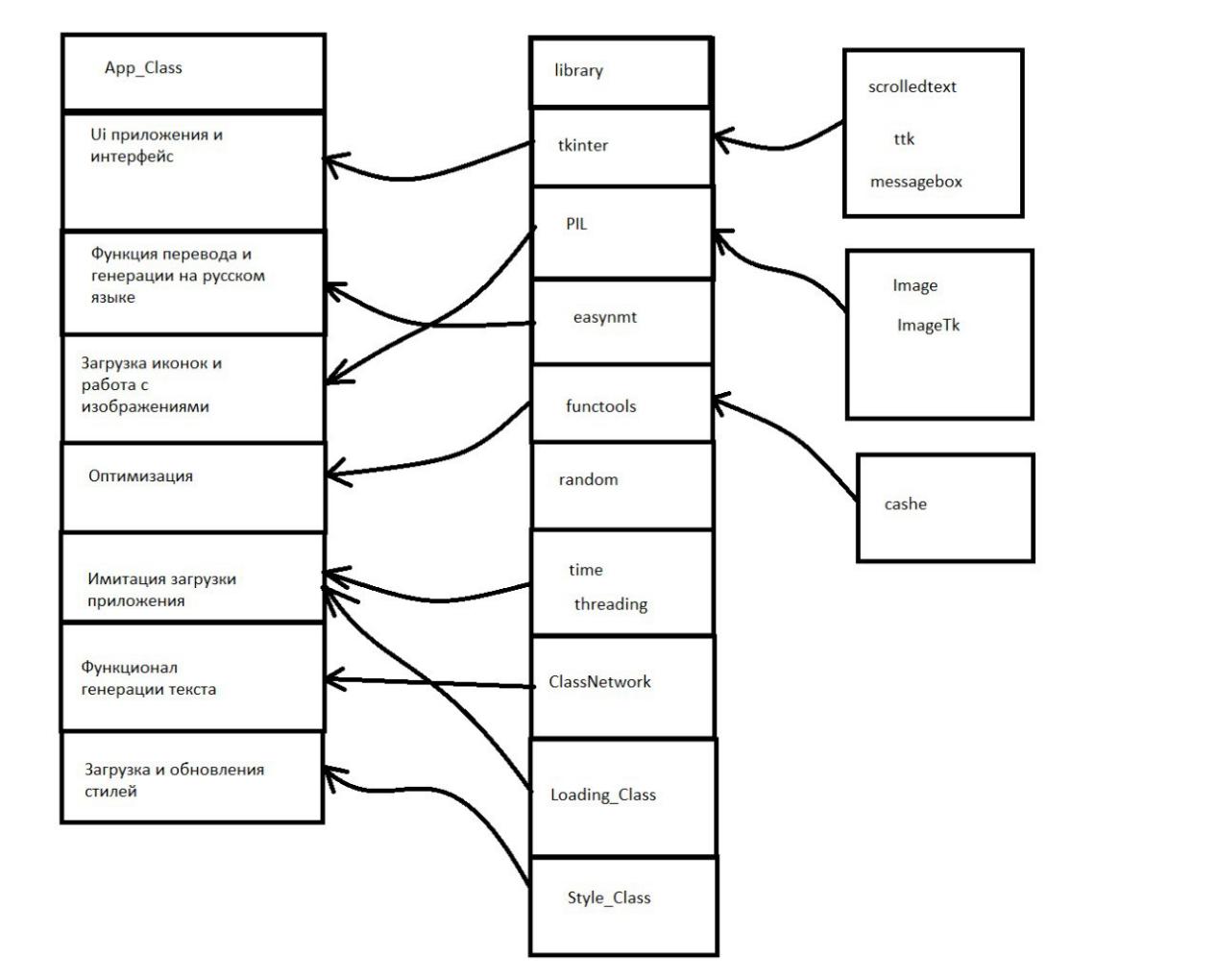
В целом концепция изменилась не сильно: также есть отдельный компонент для инициализации, далее выделены две основные подсистемы. После запуска приложения первым начинает работать компонент инициализации: происходит нажатие на кнопку и генерация текста, а также действия в зависимости от генерации текста.

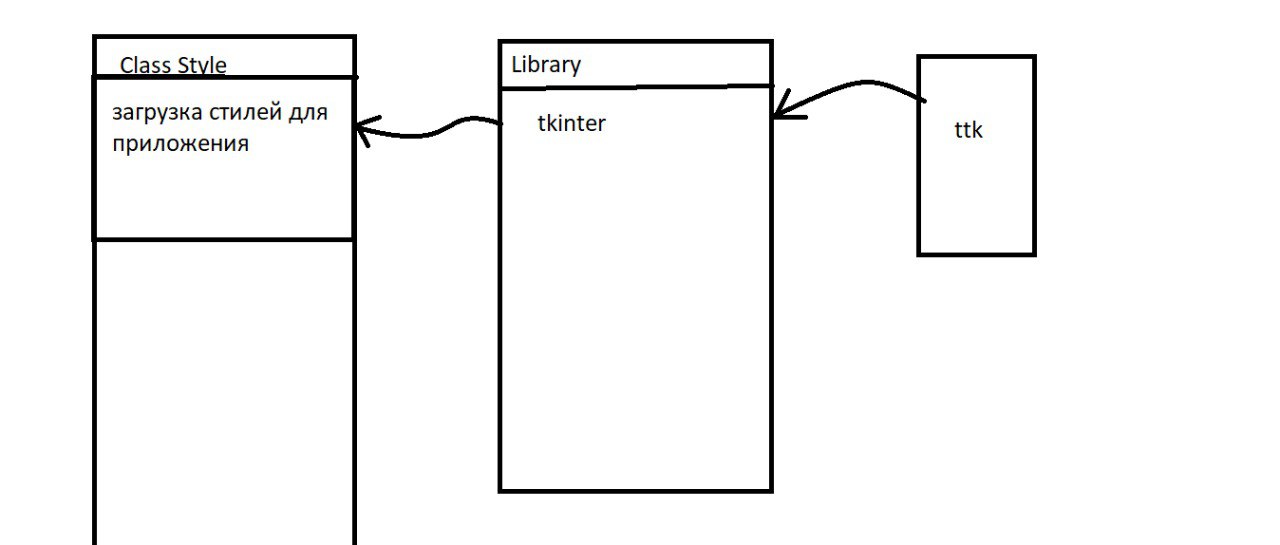
Этап 3. Фактический результат

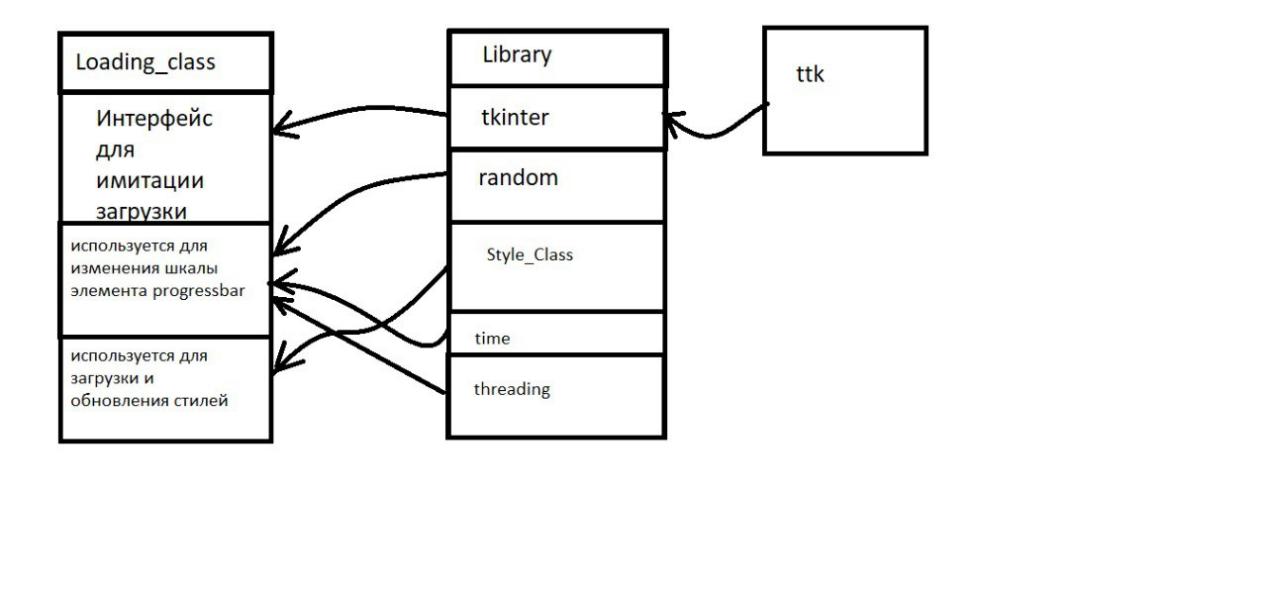
Архитектура нейронной сети разрабатывалась совместно с ней, что позволило нам сократить время на разработку архитектуры и позволило сосредоточится на нейронной сети. На диаграмме ниже представлены все классы и библиотеки, которые были созданы и использованы в разработке.

На диаграмме ниже представлены все классы, созданные при разработке нейронной сети. На диаграмме изображены все классы и её связи.









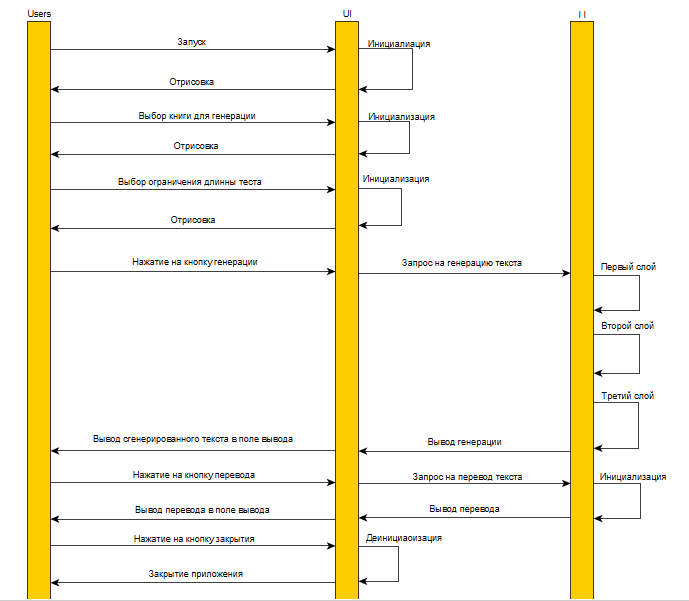
Все имеющиеся классы (библиотеки) разделены на три отдельные, но связанные между собой пакеты:

* Main - содержит в себе весь пакет библиотек, которые использовались в разработке и были разделены на сферы влияния и использования
* Библиотеки - иллюстрация всех библиотек использованных в нейронной сети
* Неименованный пакет - данный пакет является составляющей библиотеки keras

Диаграммы в свою очередь разделены на три части - Нейронная сеть, Приложение, Функция загрузка.

Этап 4. Запуск приложения

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



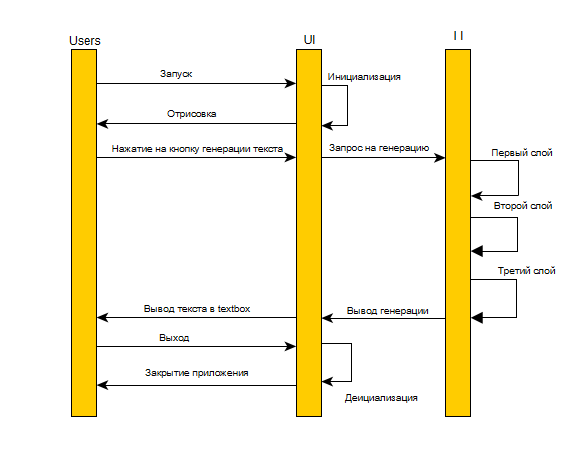
Загрузка нейронной сети при открытии приложения, также после открытия приложения пользователь нажимает на кнопку «Сгенерировать» после чего происходит генерация текста на основе обученной на книгах нейронной сети и выводит сгенерированный текст.

**Заключение**

За рамками данной статьи остались множество споров, нервов и работы, но не смотря на это разработка нейронной сети XXX является одним из запоминающихся этапов в жизни. Но также за рамками остались кропотливый труд, поиск информации, прокрастинация и многое другое.

Основные выявленные проблемы:

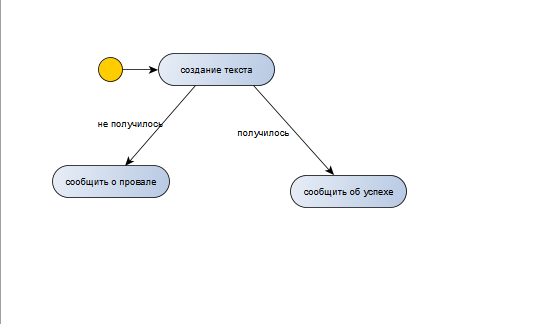
* Обучение нейронной сети
* Сохранение прогресса обучения нейронной сети



Загрузка приложения происходит при открытии приложения, а также после открытия приложения пользователь выбирает книгу, на основе которой будет генерироваться текст, затем пользователь выбирает количество символов и нажимает на кнопку «Сгенерировать текст» после чего происходит генерация текста на основе выбранной книги и размера текста и вывод сгенерированного текста в поле вывода.

Этап 5. Запуск нейронной сети

Далее рассмотрим диаграмму последовательности, иллюстрирующую действия при запуске приложения.



При запуске приложения идёт генерация текста, в случае если генерация успешна, то вывести сообщение об успехе в обратном случае сообщить о провале.

Заключение

За рамки данной статьи достаточно много