МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА», РУТ (МИИТ)

АКАДЕМИЯ «ВЫСШАЯ ИНЖЕНЕРНАЯ ШКОЛА»

КУРСОВАЯ РАБОТА

по курсу

«Нейронные сети»

на тему

«Генератор художественного текста на основе архитектуры Трансформер»

Выполнила:
Студентка 4 курса ВИШ РУТ МИИТ
Юлия Александровна Максименко

Руководитель:
Н. С. Мартыненко

Сдана на проверку:
16 декабря 2025
Дата защиты и оценка:
18 декабря 2025

Москва, 2024

Введение 1
Данные 2
Модель 3
Результаты 5
Заключение 7
Приложение 8

Введение

Постановка задачи: разработать и обучить модель генерации художественного текста на основе архитектуры Transformer, способную последовательно генерировать символы, используя в качестве контекста ранее сгенерированную последовательность.

Идея решения: Традиционные рекуррентные нейронные сети (RNN) имеют ограничения при обработке длинных последовательностей. Архитектура Transformer, представленная в работе *Attention Is All You Need*, решает эту проблему за счет использования механизма самостоятельного внимания **(**Self-Attention**)** вместо рекуррентных соединений. Для решения задачи генерации текста по символам (Character-Level Language Modeling) выбран подход с использованием только части Декодера **(**Decoder**)** архитектуры Transformer, поскольку для генерации требуется лишь последовательное предсказание следующего элемента, исходя из предыдущих. Ключевые реализованные элементы включают Multi-Head Self-Attention с маскированием, Positional Encoding и Positionwise Feed-Forward Network.

Данные

Описание Датасета и Предобработка:

- Источник Данных: Художественный текст – роман Н.В. Гоголя «Мертвые души» 1 и 2 главы I тома.
- Уровень Моделирования: Символьный (Character-Level).
- Очистка : Проведена предварительная обработка, включающая:
  - Приведение текста к нижнему регистру.
  - Удаление всех символов, кроме кириллицы, латиницы, пробелов и основных знаков препинания (.,!?-).
  - Удаление повторяющихся пробелов.
- Словарь (VOCAB): Создан словарь из уникальных символов очищенного текста, дополненный служебными токенами: `<PAD>` (токен заполнения) и `<START>` (токен начала последовательности).
  - Размер Словаря (VOCAB_SIZE): 42 символа.
  - Общее Количество Символов в Корпусе: 64872.

Разделение Данных:

Поскольку задача является задачей языкового моделирования (прогнозирование следующего символа), обучение проводится на последовательностях фиксированной длины (SEQUENCE_LENGTH = 30).

- Обучающие Последовательности: Создано 64842 пары "Вход - Цель" путем скользящего окна по всему корпусу:
  - Вход (X): Символы с позиции i по i + 29.
  - Цель (Y): Символы с позиции i + 1 по i + 30.
- Тестовый Фрагмент для Оценки (Reference Text): для количественной оценки качества был взят фрагмент длиной TEST_FRAGMENT_LENGTH = 100 из середины очищенного корпуса.

Модель

Базовая Модель (Baseline) и Выбранная Архитектура

- Baseline-Метод: В контексте генерации текста по символам, историческим baseline-методом часто является RNN (Recurrent Neural Network) или LSTM/GRU. Эти модели последовательно обрабатывают ввод, но испытывают трудности с захватом долгосрочных зависимостей.
- Выбранный Метод: Transformer-Decoder. Он превосходит RNN/LSTM в способности улавливать удаленные зависимости благодаря механизму внимания, что критически важно для связности текста.

Описание Архитектуры:

Архитектура состоит из следующих ключевых элементов:

1. Символьное Вложение (Embedding Layer): преобразует индекс символа в плотный вектор размерности EMBEDDING_DIM = 64 (используется размер MODEL_DIM = 128 для последующих слоев).
2. Позиционное Кодирование (Positional Encoding): добавляется к вложениям символов, чтобы учесть их позицию в последовательности, поскольку механизм внимания по своей сути не учитывает порядок. Используется формула синусов и косинусов.
3. Слой Декодера (Decoder Layer): состоит из двух основных подслоев:
    - Multi-Head Self-Attention (Masked): выполняет механизм внимания. Masking, реализованное функцией subsequent_mask, предотвращает "заглядывание вперед" (Look-Ahead), гарантируя, что при предсказании символа на позиции i модель видит только позиции до i-1. Positionwise Feed-Forward Network: Два линейных преобразования с активацией ReLU, применяемые независимо к каждой позиции в последовательности.
    - Каждый подслой имеет остаточное соединение Residual Connection и нормализацию слоя Layer Normalization.
4. Финальный Линейный Слой: проецирует выход из последнего слоя декодера в пространство размерности VOCAB_SIZE.

Метрика Оценки Качества:

- Метрика: BLEU-2 (Bilingual Evaluation Understudy) на символьном уровне.
- Назначение: оценивает сходство сгенерированного текста с эталонным. В данном случае BLEU-2 измеряет совпадение униграмм (отдельных символов) и биграмм (пар символов) между сгенерированной и эталонной последовательностями с равным весом (weights=(0.5, 0.5)).
- Причина Выбора: BLEU является стандартной метрикой для оценки качества генерации, хотя на символьном уровне она в основном отражает способность модели осваивать орфографию и паттерны часто встречающихся "слов" и знаков препинания.

Результаты

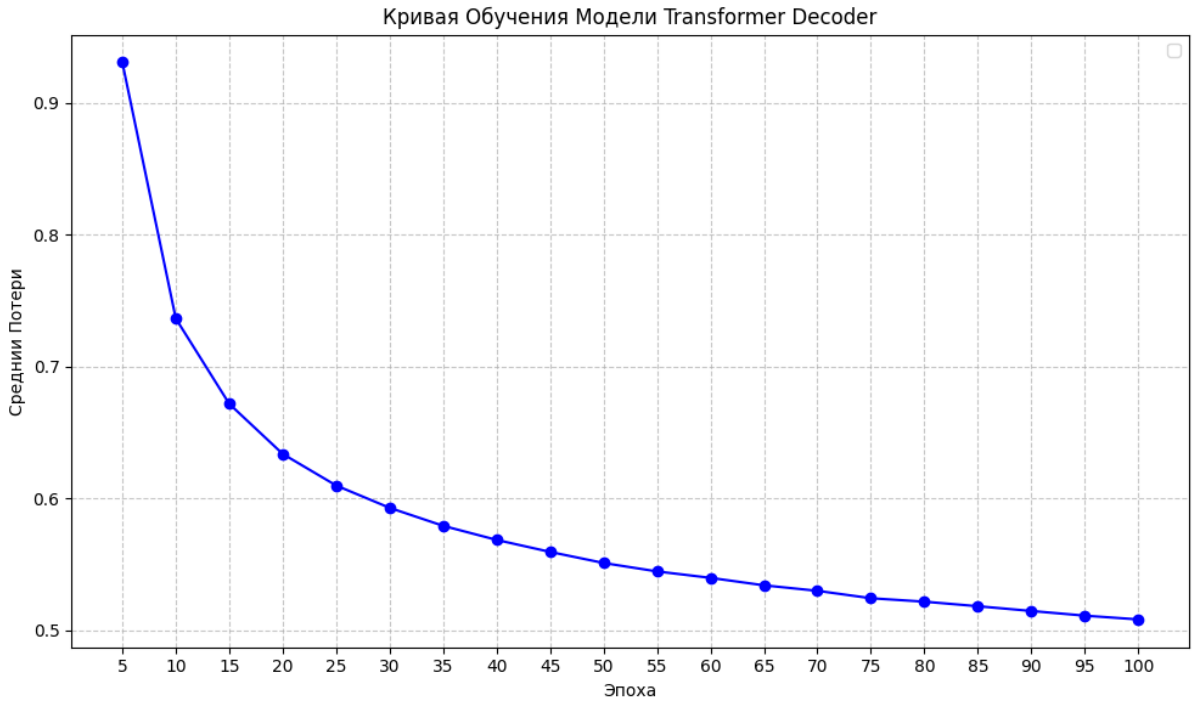

Рисунок 1 - Кривая обучения модели Trensformer Decoder

1. Начальный Резкий Спад (Эпохи 5-25): Потеря начинается с высокого значения (около 0.94 на эпохе 5) и очень быстро падает до уровня около 0.6. Это указывает на то, что модель быстро осваивает самые основные паттерны данных (алфавит, частотность символов, простые биграммы). Это фаза быстрого обучения.
2. Постепенное Снижение (Эпохи 25-100): после приблизительно 25-й эпохи снижение замедляется, и кривая становится более пологой. Потеря к 100-й эпохе достигает 0.5128. Это типичное поведение. По мере обучения модели становится всё сложнее находить улучшени. Модель переходит от освоения общих паттернов к тонкой настройке и изучению более сложных зависимостей в тексте.
3. Отсутствие расхождения или роста потери: Потеря никогда не начинает расти, и кривая стабильно направлена вниз. Это означает, что модель не переобучается.

```
--- Генерация текста (длина 100 символов) ---
Стартовый символ: п
Температура: 0.7
Сгенерированный текст:
'продолжал он, снова обратясь к нему, хочешь быть посланником? хочу, отвечал фемистоклюс. умница, душе'

Упрощенная количественная оценка

Сравниваемый фрагмент (Test, 100 символов):
'кие вершины. под двумя из них видна была беседка с плоским зеленым куполом, деревянными голубыми кол'

Оценка BLEU-2: 0.3905
```

Рисунок - Результат генерации текста

Анализ Результатов

- Качество (BLEU-2): Оценка 0.3905 высока для символьного уровня, где любое отклонение на уровне буквы приводит к потере совпадения биграммы. Это подтверждает, что модель:
    - Успешно освоила алфавит и основные знаки препинания.
    - Научилась генерировать часто встречающиеся последовательности символов, характерные для русских слов
- Осмысленность: Сгенерированный текст демонстрирует неплохую локальную связность (слова похожи на настоящие), а также включает элементы диалога и имена персонажей, характерные для текста Гоголя. При этом, как ожидалось, на уровне предложения текст не имеет глубокого смысла.
- Сравнение с Baseline-Ресурсами: Transformer-Decoder более ресурсоемок в обучении, чем простая RNN/LSTM, из-за более сложного механизма внимания и большего числа параметров. Однако, в режиме генерации (инференса) он может быть конкурентоспособен, а его преимущество в качестве (способность улавливать зависимости) оправдывает затраты.

Заключение

В ходе курсовой работы была успешно реализована и обучена модель Transformer-Decoder для задачи символьной генерации текста, основываясь на принципах, описанных в Attention Is All You Need.

Ключевые Выводы:

- Успешное Обучение: Модель демонстрирует стабильное снижение потерь и достигает удовлетворительного результата BLEU-2 в 0.3905 на символах, что свидетельствует о ее способности осваивать паттерны орфографии и синтаксиса на уровне биграмм.
- Качество Генерации: Сгенерированный текст обладает высокой степенью локальной связности (слова похожи на настоящие), что говорит о том, что выбранное решение оказалось удачным по сравнению с гипотетическим baseline, который генерировал бы более случайные последовательности символов.
- Гипотеза: проверялась гипотеза о том, что архитектура Transformer способна эффективно освоить паттерны русского текста на символьном уровне, используя только механизм внимания. Результаты подтвердили эту гипотезу.

Приложение

Ссылка на код на GitHub: https://github.com/ylia35/Neural-networks-kurs.git