



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени
Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 5 по дисциплине «Основы искусственного интеллекта»

Тема Естественные языки

Студент Сапожков А.М.

Группа ИУ7-13М

Преподаватель Строганов Ю.В.

Москва, 2024

Содержание

ВВЕДЕНИЕ	4
1 Аналитическая часть	5
1.1 Типология порядка слов	5
1.2 Цепи Маркова и n-граммы	5
2 Формирование n-грамм с помощью цепей Маркова	6
2.1 Средства реализации	6
2.2 Реализация алгоритма	6
2.3 Оценка «человечности» сгенерированных текстов	6
2.4 Работа с нестрогим порядком слов	9
3 Генерация текстов при помощи предобученных моделей	11
ЗАКЛЮЧЕНИЕ	13
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	14

ВВЕДЕНИЕ

Генерация текстов на естественном языке является актуальной задачей в области искусственного интеллекта. В современных приложениях ИИ, таких как чат-боты, системы рекомендации и переводчики, генерация текстов часто используется для улучшения пользовательского опыта взаимодействия с разрабатываемым ПО.

Одним из методов генерации текстов является использование цепей Маркова. Цепи Маркова — это математическое представление вероятностной модели последовательности случайных процессов, которые могут быть применены к генерации текста на основе его контекста и соседних слов.

Целью данной лабораторной работы является изучение потенциала использования цепей Маркова для генерации текстов на естественном языке.

Задачи данной лабораторной работы:

1. Сформировать n -граммы для генерации с помощью цепи Маркова на основе обучающего текста.
2. Сформировать тексты с различными затравочными начальными словами.
3. Дать экспертную оценку «человечности» сформированных текстов.
4. Провести исследование возможности генерации текста при наличии обучающей выборки, состоящей только из предложений «кошка съела мышку» и «мышку съела кошка», оценить опасность работы с языками с нестрогим порядком слов (не обязательно SVO).
5. Осуществить генерацию текстов с помощью инструмента из представленной методички. Оценить получаемый порядок слов в генерируемых предложениях для разной версии qwen2.5.

1 Аналитическая часть

1.1 Типология порядка слов

Типология порядка слов в предложении — один из методов типологической классификации языков, учитывающий базовый порядок слов в предложении: подлежащего (subject), сказуемого (verb) и дополнения (object). Современное состояние типологии базового порядка слов представлено во Всемирном атласе языковых структур. [1]

Начало современному изучению типологии базового порядка составляющих в языках мира было положено во второй половине XX века американским лингвистом Джозефом Гринбергом. Гринберг выделил шесть базовых порядков составляющих в предложении — SOV, SVO, VSO, VOS, OVS, OSV — и установил некоторые имплицативные отношения между этим и другими порядками. Так, в соответствии с описанной классификацией, русский язык имеет базовый порядок слов SVO. В современной лингвистике базовый порядок слов в предложении не считается достаточным для типологической классификации порядка слов в языках мира и определяющим все частные порядки.

1.2 Цепи Маркова и n-граммы

Цепь Маркова — последовательность случайных событий с конечным или счётным числом исходов, где вероятность наступления каждого события зависит только от состояния, достигнутого в предыдущем событии. Характеризуется тем свойством, что её будущее состояние зависит только от текущего состояния и не зависит от предыдущих.

N-граммы – это статистические модели, которые предсказывают следующее слово после n-1 слов на основе частоты их сочетания в обучающей выборке.

С помощью цепей Маркова можно рассчитать вероятность появления следующего элемента последовательности (например, n-го слова после n-1-граммы), не учитывая слишком широкий контекст.

Вывод

В данном разделе была описана типология порядка слов в предложениях на естественных языках, а также были рассмотрены понятия Марковских цепей и n-грамм.

2 Формирование n-грамм с помощью цепей Маркова

2.1 Средства реализации

В качестве языка программирования для реализации выбранных алгоритмов был выбран язык программирования Python [2] ввиду наличия библиотек, реализующих цепи Маркова. Обучение модели для генерации текстов проводилось с использованием библиотеки markovify [3].

2.2 Реализация алгоритма

На листинге 2.1 представлена реализация алгоритма обучения модели, генерации текстов на основе затравочного слова и подсчёта частоты генерации словосочетаний.

Листинг 2.1 — Алгоритм генерации предложений на основе обучающей выборки и затравочного слова

```
from collections import defaultdict
import markovify

start_str = "Доказательство"
sentences_count = 1000

with open("texts/Кнут-том-1.txt") as f:
    text = f.read()
    for size in [1, 2, 4]:
        print(f'\nn = {size+1}:')
        sentences = defaultdict(int)
        text_model = markovify.Text(text, state_size=size)
        for i in range(sentences_count):
            sentence = text_model.make_sentence_with_start(start_str,
                                                            strict=False, tries=100)
            sentences[sentence] += 1

    print(f'generated {len(sentences)} different sentences')

    for sentence in sorted(sentences, key=sentences.get, reverse=True)
        [:5]:
        print(f'{sentences[sentence]/sentences_count:.3f}:\t {sentence}')
```

2.3 Оценка «человечности» сгенерированных текстов

Для оценки «человечности» генерируемых текстов был проведён опрос экспертов: Кириченко С. П. и Царева А. А. В таблице 2.1 представлена экспертная оценка «человечности»

текстов, сгенерированных на основе затравочного слова «Доказательство». Заметим, что для $n=5$ на основе обучающей выборки сгенерировать тексты для выбранного затравочного слова не удалось.

Таблица 2.1 — Экспертная оценка «человечности» текстов, сгенерированных на основе затравочного слова «Доказательство»

n	Предложение	Экспертная оценка	
		Кириченко С. П.	Царев А. А.
2	Доказательство данного дерева.	-	-
	Доказательство корректности алгоритма прекращается.	+	+
	Доказательство корректности алгоритма заканчивается.	+	+
	Доказательство данного корня.	-	-
	Доказательство данного раздела.	-	+
3	Доказательство проводим индукцией по n .	+	+
	Доказательство корректности алгоритма имеет и другой, еще более удачное решение, которое приведено ниже.	-	+
	Доказательство в таком порядке.	-	-
	Доказательство того факта, что частота каждой конкретной даты пасхи для заданного года, предполагая, что номер этого года не превышает количества действий S при их считывании слева направо.	-	-
	Доказательство того факта, что частота каждой конкретной даты пасхи для заданного года, предполагая, что номер этого года не превышает 100000.	-	-
5	Доказательство проводим индукцией по m ; при $m = 1$ и $n = 2$ выполнение алгоритма никогда не закончится.	+	+

В таблице 2.2 представлена экспертная оценка «человечности» текстов, сгенерированных на основе затравочного слова «Свойство». Заметим, что для $n=5$ на основе обучающей выборки сгенерировать тексты для выбранного затравочного слова не удалось.

Таблица 2.2 — Экспертная оценка «человечности» текстов, сгенерированных на основе затравочного слова «Свойство»

n	Предложение	Экспертная оценка	
		Кириченко С. П.	Царев А. А.
2	Свойство ортогональности перестановок, расположенных подряд единиц.	-	-
	Свойство ортогональности перестановок, 199, 201, 211, Харари Ф.	-	+
	Свойство ортогональности перестановок, 199, 210.	+	+
	Свойство ортогональности перестановок, 199, 206, 215, 217.	+	+
	Свойство A обозначений.	+	-
3	Свойство A всегда справедливо для конечных ориентированных графов.	+	+
	Свойство A всегда справедливо для операций сложения и умножения для списков введена Д.	-	-
	Свойство A всегда справедливо для операций сложения и умножения для списков любого типа будут практически одинаковы.	-	-
	Свойство A всегда справедливо для конечных множеств, причем упомянул, что он соответствует показанному справа тетрадному типу, в четырех различных списках.	-	-
	Свойство A всегда справедливо для всех комплексных х.	+	+

В таблице 2.3 представлена экспертная оценка «человечности» текстов, сгенерированных на основе затравочного слова «программа». Заметим, что для $n=5$ на основе обучающей выборки сгенерировать тексты для выбранного затравочного слова не удалось.

Таблица 2.3 — Экспертная оценка «человечности» текстов, сгенерированных на основе затравочного слова «программа»

n	Предложение	Экспертная оценка	
		Кириченко С. П.	Царев А. А.
2	программа поиска, 453.	+	+
	программа работает программа.	-	-
	программа вычисления, 180.	+	+
	программа была использована Й.	-	-
	программа вычисления, ввод, 254.	-	-
3	программа имеет следующий вид.	+	+
	программа является неполной в нескольких словах блока из имеющихся в настоящий момент в системе.	-	-
	программа М является искомым значением.	-	-
	программа переходит к ячейке ENTER.	+	-
	программа в процессе внедрения Scrum.	-	-

Средняя «человечность» сгенерированных текстов составила 0.5 для $n=2$, 0.33 для $n=3$ и 1 для $n=5$. Для выявления корреляции между n и «человечностью» текстов следует провести подобное исследование на большем массиве данных.

2.4 Работа с нестрогим порядком слов

Для оценки возможности генерации текстов при помощи цепей Маркова на основе обучающей выборки, содержащей предложения с разным базовым порядком слов была взята выборка из предложений «кошка съела мышку» и «мышку съела кошка».

На листинге 2.2 представлены результаты генерации тестов на основе описанной обучающей выборки для затравочного слова «кошка».

Листинг 2.2 — Наиболее частые сгенерированные словосочетания и их частота

```
generated 2926 different sentences
0.333:   кошка съела кошка
0.084:   кошка съела кошка съела мышку
0.083:   кошка съела кошка съела кошка
0.083:   кошка съела мышку съела кошка
0.083:   кошка съела мышку съела мышку
0.021:   кошка съела кошка съела мышку съела мышку
0.021:   кошка съела кошка съела кошка съела кошка
0.021:   кошка съела мышку съела мышку съела мышку
```


0.021:	кошка съела мышку съела кошка съела мышку
0.021:	кошка съела кошка съела мышку съела кошка

Ни одно из сгенерированных словосочетаний эксперты не сочли «человечным».

Вывод

В данном разделе были описаны детали реализации алгоритма генерации тестов, дана экспертная оценка «человечности» сгенерированных текстов и продемонстрирована работа с нестрогим порядком слов. При работе с обучающей выборкой, имеющей нестрогий базовый порядок слов, модель на основе цепей Маркова «путается», закливая словосочетания и полностью искажая их смысл, содержащийся в обучающей выборке. Для избежания описанной проблемы предлагается использовать обучающие выборки большего объёма, а также фиксировать базовый порядок слов в предложениях.

3 Генерация текстов при помощи предобученных моделей

Для генерации текстов были взяты предобученные модели qwen2.5 [4] с тегами 0.5b, 7b и 14b, означающими число параметров модели. Для оценки «человечности» текстов, сгенерированных моделями, была взята затравочная фраза с нестрогим порядком слов: «Кошка съела мышку? Или мышку съела кошка? А может кошка мышку съела? Или всё-таки съела кошка мышку?». Тексты, сгенерированные рассматриваемыми моделями, представлены на листингах 3.1-3.3.

Листинг 3.1 — Текст сгенерированный моделью qwen2.5:0.5b

Что бы вы хотели сказать, это решает 2 вещи:

1. Мыши могут съесть кошек или кошек тоже могут съесть мышей.
2. С другой стороны, кошки не могут съесть мышинного животного.

Эти два вопроса имеют одинаковую ответную форму - "или всё-таки съела кошка мышку? Или кошка мышка съела кошки?"

Но, к сожалению, даже с учетом этих обстоятельств, можно сказать следующее:

1. Мыши могут съесть кошек.
2. Кошки могут съесть мышинного животного.

Изначально эта ситуация может показаться непонятной, но в реальности она действительно существует и работает.

Однако, давайте попробуем разобрать её более подробно:

1. ****Мыши могут съесть кошек**:**

- Кошки - это животные, которые обычно не съедают людей.
- Мышки - это животные, которые часто пишут и часто едят мускулатуру.

2. ****Кошки могут съесть мышинного животного**:**

- Мускулатура - это съедобная часть человека, включая кожу, ртушину, желтый оттенок кожи и даже мочи.
- Мыши - это животные, которые часто пишут, что указывает на то, что они также могут съесть мускулатуру.
- Мыши - это животные, которые обычно не едят людей.

3. ****С другой стороны, кошки не могут съесть мышинного животного**:**

- Кошки - это животные, которые часто пишут и ездят в поездке.

Таким образом, в целом ситуация может показаться непонятной из-за того, что мыши могут съесть кошек, но кошки не могут съесть мускулатуру.

Листинг 3.2 — Текст сгенерированный моделью qwen2.5:7b

Правильный порядок слов в последнем варианте будет: "Кошка съела мышку".

Это предложение грамматически правильное и ясно передает действие.

В других вариантах порядок слов не соответствует русской грамматике:

1. "Кошка съела мышку? это вопрос, который можно использовать для проверки понимания.
2. "Мышку съела кошка? тоже вопрос, но менее естественный, чем первый вариант.
3. "Кошка мышку съела грамматически некорректно, так как местоимение должно стоять после глагола.

Таким образом, правильный ответ: кошка съела мышку.

Листинг 3.3 — Текст сгенерированный моделью qwen2.5:14b

Все эти варианты передают одну и ту же идею, но наиболее правильным и распространённым в русском языке будет вариант: "Кошка съела мышку." Однако, варианты "Мышку съела кошка" и "Кошка мышку съела" тоже корректны, просто они менее естественны для употребления. Важно отметить, что порядок слов в русском языке может меняться для акцентирования внимания на определённые части предложения или по стилистическим соображениям, но смысл при этом остаётся неизменным.

По мнению экспертов, текст, сгенерированный первой моделью, нельзя считать «человечными» как минимум потому, что сгенерированный текст даёт неестественный ответ на поставленный вопрос. Во втором тексте была замечена оценка естественности подвопросов исходного вопроса, но в данном тексте также присутствует ошибка разбора грамматики исходного вопроса. Третий текст содержит разбор исходного запроса, а также, по мнению экспертов, может считаться «человечным».

Также стоит отметить, что во всех сгенерированных текстах прослеживается строгий порядок слов SVO. Исходя из этого, можно сделать предположение о том, что обучающая выборка для модели qwen имела строгий базовый порядок слов в предложениях, а именно SVO.

Вывод

В данном разделе была рассмотрена генерация текстов при помощи различных версий предобученной модели qwen. На основе сгенерированных текстов можно сделать вывод о том, что между «человечностью» текстов и числом параметров генерирующей модели существует прямая зависимость.

ЗАКЛЮЧЕНИЕ

В рамках лабораторной работы был изучен потенциал использования цепей Маркова для генерации текстов на естественном языке. Все поставленные задачи были выполнены.

1. Сформированы n-граммы для генерации с помощью цепи Маркова на основе обучающего текста.
2. Сформированы тексты с различными затравочными начальными словами.
3. Дана экспертная оценка «человечности» сформированных текстов.
4. Проведено исследование возможности генерации текста при наличии обучающей выборки, состоящей только из предложений «кошка съела мышку» и «мышку съела кошка», оценена опасность работы с языками с нестрогим порядком слов (не обязательно SVO).
5. Осуществлена генерация текстов с помощью инструмента из представленной методички. Оценён получаемый порядок слов в генерируемых предложениях для разной версии qwen2.5.

При работе с обучающей выборкой, имеющей нестрогий базовый порядок слов, модель на основе цепей Маркова «путается», закликивая словосочетания, полностью искажая их смысл, содержащийся в обучающей выборке. Для избежания описанной проблемы предлагается использовать обучающие выборки большего объёма, а также фиксировать базовый порядок слов в предложениях.

При изучении текстов, сгенерированных предобученной моделью qwen2.5 было замечено, что «человечность» текстов напрямую зависит от числа параметров модели. Также было выдвинуто предположение о том, что обучающая выборка для модели qwen имела строгий базовый порядок слов в предложениях, а именно SVO.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Feature 81A: Order of Subject, Object and Verb [Электронный ресурс]. — Режим доступа, URL: <https://wals.info/feature/81A#2/18.0/152.8> (дата обращения: 20.11.2024).
2. Python [Электронный ресурс]. — Режим доступа, URL: <https://www.python.org/> (дата обращения: 19.11.2024).
3. Markovify [Электронный ресурс]. — Режим доступа, URL: <https://pypi.org/project/markovify/> (дата обращения: 20.11.2024).
4. qwen2.5 [Электронный ресурс]. — Режим доступа, URL: <https://ollama.com/library/qwen2.5> (дата обращения: 20.11.2024).