

## Вопрос 1

Выполнен

Баллов: 1,00 из 1,00

Вы делаете авторегрессионную языковую модель (то есть chain rule применяется так, чтобы всегда предсказывать следующее слово на основе фрагмента текста непосредственно слева).

Отметьте, какие обучающие примеры можно сгенерировать из текста "Наша суть отражается в наших повторяющихся действиях. Отсюда следует, что совершенство есть не действие, а привычка."

Варианты ответа приведены в формате "Вход -> Эталонный выход".

Выберите один или несколько ответов:

- ☒ a. Наша суть отражается -> в
- ☒ b. Наша суть -> отражается
- ☐ c. не действие, а привычка -> совершенство
- ☒ d. Наша -> суть
- ☐ e. Отсюда следует, что -> привычка

Ваш ответ верный.

## Вопрос 2

Выполнен

Баллов: 1,00 из 1,00

Допустим, предложения в некотором языке имеют среднюю длину в 5 слов. Средняя длина слова в символах - 5.

Предположим также, что для этого языка характерно наличие далёких связей между словами (кореферентность, непроективные синтаксические связи).

Тогда чтобы успешно предсказывать последнее слово в предложении, языковой модели, работающей на уровне целых слов (word-level), нужно в среднем помнить 5 предыдущих элементов, а модели на уровне символов (character-level) - 25.

Почему character-level моделям сложнее?

Отметьте один или несколько вариантов.

Выберите один или несколько ответов:

- ☒ a. Если строить авторегрессионную языковую модель с помощью рекуррентных нейросетей, то с ростом длины последовательности острее проявляется проблема затухания градиента и забывания со временем.
- ☒ b. Если строить авторегрессионную языковую модель с помощью свёрточных нейросетей, сложно учитывать широкий контекст, так как ширину рецептивного поля свёрточных нейросетей можно увеличивать только вместе с количеством параметров (добавляя слои).
- ☐ c. При переходе на уровень символов вырастает размер словаря и становится дорого считать softmax.
- ☐ d. Длинные последовательности не влезают в память видеокарт.

Ваш ответ верный.

### Вопрос 3

Выполнен

Баллов: 1,00 из 1,00

Byte Pair Encoding - простой алгоритм кодирования, который позволяет сжать длину последовательностей за счёт расширения словаря (то есть можно выбирать разные соотношения длины последовательностей и размера словаря).

Этот алгоритм работает в два этапа:

1. обучение - построение правил замены символов
2. кодирование - применение построенных правил замены, чтобы сократить длину текста

Разберём упрощённый алгоритм обучения на примере токенизации "косил косой осой".

1. Найти самую частотную бигramму - по два употребления имеют "ко", "ос", "со" и "ой" - выбираем "ко".
2. Назначить выбранной бигramме номер. Мы вместо номеров будем использовать заглавные буквы. Назначили: "ко" -> А.
3. Обновить последовательность с учётом созданного правила подстановки "ко" -> А: "Асил Асой осой".
4. Повторяем шаги 1-3 для бигramмы "со": подстановка "со" -> Б, последовательность "Асил АБй оБй"
5. Повторяем шаги 1-3 для бигramмы "Бй": подстановка "Бй" -> В, последовательность "Асил АВ оВ".
6. Останавливаемся, когда сделаем заданное количество замен (то есть когда достигнем максимально допустимого размера словаря) или когда заменами не сможем дальше сжать тексты обучающей выборки.

В результате для этой последовательности мы получили словарь размера 10 (7 исходных символов и 3 подстановки) и уменьшили её длину с 16 до 10.

Когда мы кодируем текст, мы повторяем процедуру, жадно (то есть без возвратов и перебора) выбирая самые частотные бигramмы и применяя подстановки, сохранённые на этапе обучения. Может так получиться, что две бигramмы имеют одинаковую частоту - тогда применяйте правила подстановки в том порядке, в котором они добавлялись в словарь при обучении.

Примените полученные правила подстановки для кодирования текста "косой кокос".

Выберите вариант ответа, соответствующий результату.

Выберите один ответ:

- ☐ a. АБй ААс
- ☐ b. коБй ААс
- ☒ c. АВ ААс
- ☐ d. Асой ААс

Ваш ответ верный.

#### Вопрос 4

Выполнен

Баллов: 1,00 из 1,00

Допустим, мы выбрали порядок факторизации 3 -> 1 -> 2 для предложения "мама мыла раму". Это означает, что будут формироваться следующие обучающие примеры (в формате "Вход -> Эталонный выход", в скобках после слова стоит его исходная позиция в тексте):

- <start> -> раму(3)
- <start> раму(3) -> мама(1)
- <start> раму(3) мама(1) -> мыла(2)

Так как порядок слов в исходном предложении очень важен для понимания его смысла, необходимо добавлять к эмбедингу слов эмбединг его исходной позиции в тексте. Это называется позиционным кодированием (подробнее - [в лекции](#) и [семинаре](#) про трансформер).

Отсортируйте слова предложения "на дворе трава на траве дрова" для порядка факторизации 1 -> 3 -> 4 -> 2 -> 6 -> 5 так, чтобы сверху от каждого слова были те слова, на основании которых оно должно быть предсказано (то есть вход модели).

на(1)

трава(3)

на(4)

дворе(2)

дрова(6)

траве(5)

Ваш ответ верный.