

Вопрос 1

Выполнен

Баллов: 1,00 из 1,00

Выберите правильные утверждения про маску зависимости позиций:

Выберите один или несколько ответов:

- ☒ a. количество строк и столбцов в маске совпадает
- ☐ b. маска является нижнетреугольной матрицей
- ☒ c. на пересечении строки i и столбца j в маске стоит 0, если при предсказании токена на позиции i можно учитывать токен на позиции j
- ☐ d. маска является диагональной матрицей

Ваш ответ верный.

единицы, но он их не переупорядочивает, а значит точка максимума не изменится. Вот мы выбираем лучший токен, используя всего лишь предсказание модели именно для этой позиции. Если на очередном шаге мы предсказали конец последовательности, то мы прекращаем генерировать дальше. А если это ещё не конец последовательности, то мы добавляем текущий токен к нашему списку токенов и снова засовываем это всё в модель на следующей итерации. Затем, когда мы сделали достаточное количество шагов, мы просто декодируем список номеров токенов в строку (всё просто). Давайте посмотрим, что же наша модель может генерировать. На вход модели мы подаём какой-то небольшой фрагмент текста. Хочу обратить ваше внимание, что мы подаём не только фрагменты текстов, но даже фрагменты слова. То есть — мы не полностью подали последнее слово, и модель сгенерировала на выходе вот такой фрагмент. То есть она, во-первых, закончила слово, а потом продолжила относительно связным текстом. У неё даже получилось смешать французский и русский язык. Обратите внимание, что данная реализация — она не очень эффективная, потому что она полностью перевычисляет всю модель, хотя, казалось бы, мы можем часть активаций сохранить, потому что когда мы в цикле в "greedy_generator" выбираем следующий лучший токен, мы всё равно берём все предыдущие токены, прогоняем их через модель, и активации на большом количестве слоёв — они будут те же самые. То есть, казалось бы, их можно закэшировать и не прогонять одни и те же токены через модель повторно. Просто закэшировав активации. Но реализация этого не очень очевидная, более того, она достаточно сложная, и поэтому в семинаре мы не стали её делать. Также на экране вы видите парочку других примеров, которые были сгенерированы моделью из практически того же текста, но с небольшими изменениями. Мы видим что, несмотря на то, что изменения в исходном тексте небольшие, результирующий текст кардинально меняется.

Вопрос 2

Выполнен

Баллов: 1,00 из 1,00

Только что для генерации текста мы использовали алгоритм жадного поиска. Какими свойствами, по вашему мнению, он обладает?

Выберите один или несколько ответов:

- ☒ a. При жадном поиске мы можем сгенерировать токен конца предложения слишком рано, никак это исправить уже не получится
- ☐ b. Жадный поиск позволяет одновременно оценивать вероятности для нескольких вариантов, сохраняя древовидную структуру возможных токенов
- ☒ c. Жадный поиск не гарантирует обнаружения оптимального решения: возможно, выбрав менее вероятную букву сейчас, на следующем шаге мы сможем сделать выбор так, что результирующая вероятность нежадного варианта будет выше

Ваш ответ верный.

Вопрос 3

Выполнен

Баллов: 1,00 из 1,00

Только что для генерации текста мы использовали алгоритм BEAM.
Выберите верные, по вашему мнению, утверждения об этом алгоритме.

Выберите один или несколько ответов:

- ☒ а. BEAM является компромиссом между скоростью работы и количеством найденных вариантов.
- ☐ b. BEAM работает быстрее жадного поиска.
- ☒ с. BEAM без нормализации чаще отдает предпочтение коротким последовательностям.

Ваш ответ верный.