V-gram'ы и все-все-все

И. Куралёнок

СПб, 2018

n-gram -> v-gram

Есть такая задача, последовательности анализировать, самый простой способ n-gram'ы.

- Активно применяются в текстах: fastText от FB, например
- Нужны для биологии (только там они k-меры :))
- Восточные языки, где деление на слова зависит от семантики
- etc.

Однако n-gram'ы ограничены по длине, и от этого ограничения можно избавиться. $\Rightarrow$ v-gram'ы!

Что это за зверь?

Если кратко, то берем алгоритм LZ77, разворачиваем окно на бесконечность, получаем словарь n-gram, которые и называем дальше v-gram.
Что с этим добром можно делать:

- Сжимать, например выдача Яндекс сжимается так быстрее чем gzip и 4 раза лучше (используется в приложении Яндекс)
- Использовать полученные v-gram'ы как слова и применять стандартные методы анализа текста
- Изучать последовательности с помощью словаря. Например магический смысл последовательности: AGTAATGGGATGGCTGGGTCAAATGGTATTTCTAGTTCTAGAT

Что нужно делать

- Попробовать обогнать Mikolov'а и со.:
  - Провести эксперименты по сжатым текстам: натравливаем v-gram'ы, применяем SVM смотрим что получилось
  - Сделать классический синонимический embedding по wikipedia
- Исследовать переносимость v-gram с одной коллекции на другую, близкую по теме
- Переписать это добро на C++ и сделать биндинг в Python

Что для этого нужно

- Знать Java, местами она довольно суровая
- Хотя бы приблизительно понимать слова постановки
- Математика на уровне "что такое кросс-энтропия без википедии"
- Приходить в Таймс примерно 1 раз в неделю и фигачить $>$ 4-х часов в неделю