

Продвинутое машинное обучение:

Домашнее задание 3

Третье домашнее задание посвящено достаточно простой, но, надеюсь, интересной задаче, в которой потребуются творчески применить методы сэмплирования. Как и раньше, в качестве решения **ожидается ссылка на jupyter-ноутбук на вашем github (или публичный, или с доступом для snikolenko); ссылку обязательно нужно прислать в виде сданного домашнего задания на портале Академии**. Как всегда, любые комментарии, новые идеи и рассуждения на тему категорически приветствуются.

В этом небольшом домашнем задании мы **попробуем улучшить метод Шерлока Холмса**. Как известно, в рассказе *The Adventure of the Dancing Men* великий сыщик расшифровал загадочные письма, которые выглядели примерно так:



Пользовался он для этого так называемым частотным методом: смотрел, какие буквы чаще встречаются в зашифрованных текстах, и пытался подставить буквы в соответствии с частотной таблицей: Е — самая частая и так далее.

В этом задании мы будем разрабатывать более современный и продвинутый вариант такого частотного метода. В качестве корпусов текстов для подсчётов частот можете взять что угодно, но для удобства вот вам “Война и мир” по-русски и по-английски:

<https://www.dropbox.com/s/k23enjvr3fb40o5/corpora.zip>

1. Реализуйте базовый частотный метод по Шерлоку Холмсу:
 - подсчитайте частоты букв по корпусам (пунктуацию и капитализацию можно просто опустить, а вот пробелы лучше оставить);
 - возьмите какие-нибудь тестовые тексты (нужно взять по меньшей мере 2-3 предложения, иначе вряд ли сработает), зашифруйте их посредством случайной перестановки символов;
 - расшифруйте их таким частотным методом.
2. Вряд ли в результате получилась такая уж хорошая расшифровка, разве что если вы брали в качестве тестовых данных целые рассказы. Но и Шерлок Холмс был не так уж прост: после буквы Е, которая действительно выделяется частотой, дальше он анализировал уже конкретные слова и пытался угадать, какими они могли бы быть. Я не знаю, как запрограммировать такой интуитивный анализ, так что давайте просто сделаем следующий логический шаг:
 - подсчитайте частоты *биграмм* (т.е. пар последовательных букв) по корпусам;

