Advanced Machine Learning: HW 3

https://docs.google.com/document/d/1FWCuz-3Q_85yQYEwz6xVkIuXUjQn60dVFxreJId-liM

The third homework assignment focuses on a fairly simple, but hopefully interesting task in which you will need to creatively apply sampling techniques. As before, a link to the jupyter notebook on your github (either public or with access for snikolenko) is expected as a solution; the link must be sent in the form of the completed homework on the Academy's portal. As always, any comments, new ideas and reflections on the topic are strongly welcomed.

In this little homework assignment, we will try to improve the Sherlock Holmes method. As you know, in the story The Adventure of the Dancing Men, the great detective deciphered mysterious letters that looked something like this:

For this, he used the so-called frequency method: he looked which letters are more common in cipher texts, and tried to substitute the letters in accordance with the frequency table: E is the most frequent, and so on.

In this assignment, we will develop a more modern and advanced version of this frequency-based method. You can take anything you like as text corpora for calculating frequencies, but for convenience, here's “War and Peace” in Russian and in English:

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

1. Implement the Sherlock Holmes Basic Frequency Method:

    - count the frequencies of letters by corpus (punctuation and capitalization can be simply omitted, but it is better to leave spaces);
    - take some test texts (you need to take at least 2-3 sentences, otherwise it is unlikely to work), encrypt them by randomly rearranging characters;
    - decode them using this frequency method.

2. It is unlikely that the result was such a good transcript, except if you took whole stories as test data. But Sherlock Holmes was not so simple either: after the letter E, which really stands out in frequency, then he analyzed specific words and tried to guess what they could be. I don't know how to program this kind of intuitive analysis, so let's just take the next logical step:

    - count the frequencies of the bigrams (i.e. pairs of consecutive letters) by their bodies;
    - test in the same way as in point 1, but with the help of bigrams.

3. But that's not all: bigrams most likely also do not always work. The main part of the assignment is how you can improve them:

   - Suggest a method for teaching character permutation in this task, based on MCMC sampling, but still working on the basis of bigram statistics;
   - implement and test it, see if the results get better.

4. Decrypt the message:

   ← ⇠⇒↑↹↷↓↓↹↷↑↤↑↕ ← ↹↝⇛⇮↬⇒↑↑↝↓↓↾↹↑⇒↑↹↞⇛↕↑↹↝⇛⇮↬⇒↑↑↝↓↓↾↹↕ ← ⇌ ⇠↕↹↙↹↦↕⇛↙⇛↹⇠⇛⇛↲↺ ← ↝↑↞↹⇌⇛↕⇛⇮↓↓↾↹⇒ ← ↙⇌⇛↹↞⇮⇛↑↕↬↕↑↑↹⇠⇌⇛⇮ ← ← ↹↷⇠ ← ↙⇛↹↷↓↓↹↷⇠ ← ↹⇠↤ ← ⇒↬⇒↑↹↞⇮↬↷↑⇒↑↑↝⇛↹↑↹↞⇛⇒↙↟↑↕ ← ↹↳↬⇌⇠↑↳ ↬⇒↑↑↝↓↓↾↹↲↬⇒⇒↹⇨↬↹↞⇛⇠⇒ ← ↤↝ ← ← ↹↟ ← ↕↷ ← ⇮↕⇛ ← ↹⇨↬↤↬↝↑ ← ↹⇌↙⇮⇠↬↹↘ ⇛↕↞↹⇌⇛↝ ← ↟⇜⇛↹↞↹↝↑↟ ← ↙⇛↹↝ ← ↹⇛↲ ← ⇆↬≠

   Or this (they are the same, the second option is just in case of problems with unicode):