Душкин Роман Викторович
Директор по науке и технологиям искусственной интеллектуальной системы для персональной медицины «Джейн», автор курса по основам искусственного интеллекта (ai101.ru).

Биграммный шифр

Продолжаем изучение методов криптографии и криптоанализа, начатое в № 9/2017 журнала «Потенциал». На этот раз мы изучим, что такое биграммный шифр, как при помощи него снова можно удивить своих товарищей, ещё плохо разбирающихся в криптографии, а также узнаем, как его можно взломать.

В первой статье цикла мы научились придумывать шифры простой замены, зашифровывать при помощи них свои секреты и разгадывать секреты своих товарищей, которые зашифровали их простым шифром. Уверен, что в школе у вас могло произойти что-то такое. Сначала читатель зашифровал несколько своих посланий, чем удивил своих одноклассников. Потом он поделился ключом со своими друзьями, и все увидели, как это интересно. Началось повальное увлечение шифрами. Потом мой читатель атаковал несколько перехваченных шифровок и успешно дешифровал их, чем вызвал ещё большее удивление, восторг и восхищение (а, быть может, и зависть). Но потом все постепенно поняли, как это делается и что никаких тайн здесь нет. Теперь пришло время удивить всех ещё раз.

Сегодня мы изучим следующий по сложности шифр, который называется биграммным. Этот шифр может сразу же отпугнуть от попыток расшифровки даже тех, кто успешно справляется с шифром простой подстановки. Тем не менее, мы также научимся его взламывать, поскольку это совсем не сложно. Итак, поехали...

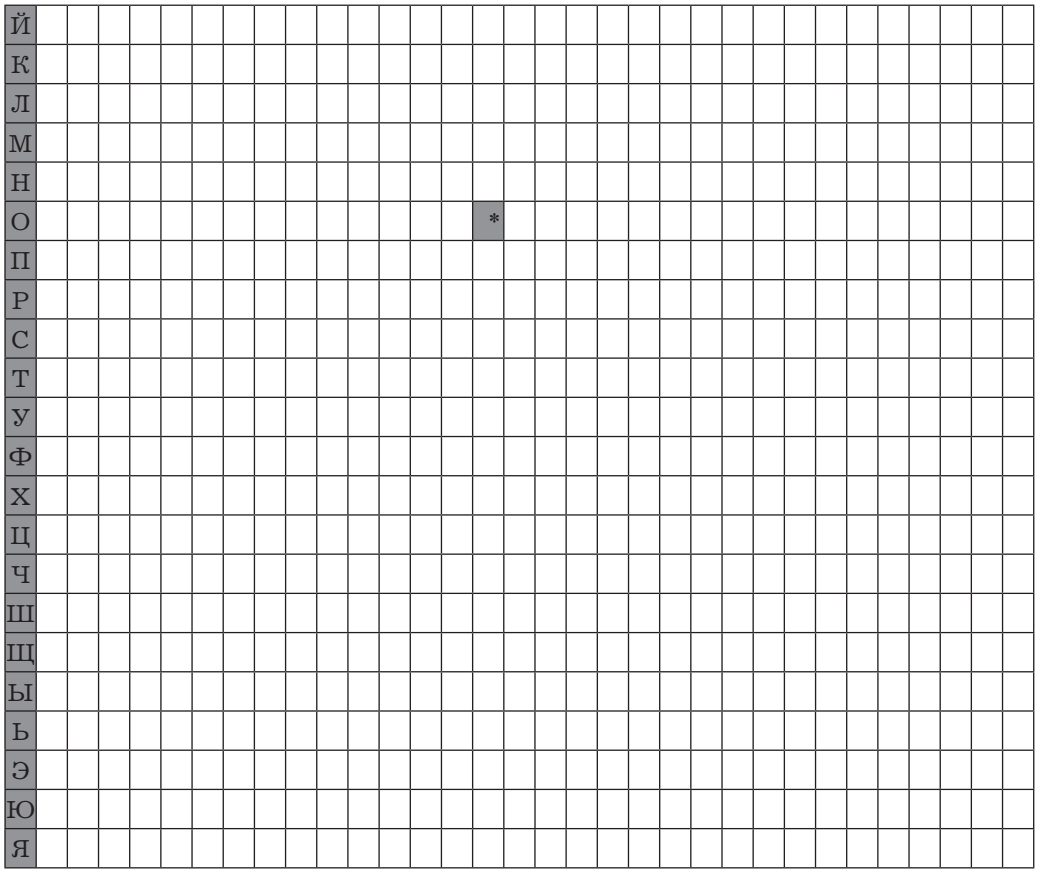
Метод шифрования

Давайте подумаем, что обозначает слово «биграммный»? Если обратиться к древним языкам – древнегреческому и латыни, то из них мы найдём ответ на этот вопрос. Вообще, было бы хорошо знать приставки и корни древнегреческих и латинских слов, чтобы сразу понимать смысл научной терминологии, ведь большинство научных терминов образовано из корней древних языков, которые раньше были языками науки и научного сообщества, позволяя учёным обмениваться знаниями независимо от национальных языков, на которых учёные говорили в быту. Так вот, приставка «би-» пришла из латыни, где слово *bis* обозначает «дважды». Помните, как в театре зрители кричат «Браво! Бис!», если хотят, чтобы представление было продемонстрировано заново? Теперь посмотрим на корень «грамма». Он пришёл из древнегреческого языка, в котором есть слово γράμμα, которое переводится просто как «буква» или же «знак», «запись». Навести на идею об этом помогут такие слова, как «анаграмма», «диаграмма», «программа» и т. д. Но теперь у нас всё есть, и мы можем ответить на вопрос, что же такое «биграммный». Биграммный – это состоящий из двух букв, символов.

Большинство корней научных терминов взяты из древнегреческого языка и латыни

Другими словами, биграммный шифр – это шифр, в котором отдельный код или тайный символ получает пара символов исходного алфавита. Давайте конкретизируем это определение, и на примере увидим, что это такое и как просто это сделать. Возьмём, как обычно, русский алфавит. Пусть для нашего примера две пары букв – Е и Ё, а также Ъ и Ь не различаются, то есть букв всего будет 31, а вместе с пробелом всего в алфавите будет 32 символа. Теперь составим матрицу 32×32, представленную ниже:

	–	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ы	Ь	Э	Ю	Я
–																																
А																																
Б																																
В																																
Г																																
Д																																
Е																																
Ж																																
З																																
И																																



Строки матрицы помечены буквами, которые стоят на первом месте биграммы, а столбцы – буквами, стоящими на втором месте. То есть, к примеру, биграмма «ОН» соответствует ячейке, выделенной в представленной матрице цветом с символом * в ней.

Поскольку матрица у нас 32×32, всего в ней получается 1024 ячейки. Это значит, что для подготовки шифра требуется придумать 1024 значка. Это очень непросто, и когда в своём отрочестве я придумал эту идею для того, чтобы опять скрывать свои секреты от одноклассников, которых я обучил частотному анализу, мне пришлось очень сильно напрягать свою фантазию и использовать все значки всех доступных мне тогда алфавитов – глаголицы, германских старших рун и скандинавских младших рун, армянского и грузинского алфавитов, семитских типов письма и многих других. Но символов всё равно не хватало, приходилось придумывать всякие вычурные значки. Чтобы этого не делать, надо просто воспользоваться числами.

Если же воспользоваться числами, то возникает интересная проблема. Необходимо выбрать 1024 различных числа, то есть каждая биграмма будет представляться четырёхзначным числом. Всего существует 10 000 четырёхзначных чисел от 0000 до 9999, так что необходимо осуществить случайную выборку без повторов 1024 из 10 000. Лучше всего это сделать при помощи про-

граммы для компьютера, так что если вы умеете программировать на каком-либо языке программирования, то напишите программу, которая составляет матрицу размером 32 × 32 и заполняет её случайными неповторяющимися числами от 0000 до 9999, при этом рекомендую все числа делать четырёхзначными с лидирующими нулями, если это необходимо. Если же вы программировать ещё не научились в том объёме, в каком это необходимо для решения поставленной задачи, то можно воспользоваться таким методом.

Десять монеток

Возьмите 10 монет и как-то пометьте их от 1 до 10. Бросайте все десять монет, а потом выстраивайте двоичное число, что-то типа 0010011001, где 0 на первом месте обозначает орла на первой монете, 0 на втором месте обозначает орла на второй монете, 1 на третьем месте обозначает решку на третьей монете и т. д. до десятой монеты. Потом переводите полученное число из двоичной в десятичную систему счисления, то есть в этом случае в 0153. Это довольно трудоёмкий процесс, поскольку вам придётся ещё проверять на дублирование, так что в какой-то момент вам захочется бросить это дело и просто напридумывать чисел. Но это неправильный шаг, так как в таком случае велика вероятность, что вы не сможете обеспечить равномерность распределения чисел в выборке, а это значит, что у криптоаналитика будут зацепки для взлома такого шифра.

Что ж, перед тем как перейти к изучению методов атаки на этот способ шифрования, рассмотрим немного исторических сведений. Поговаривают, что первым из всех людей биграммный шифр придумал бенедиктинский монах Иоганн Тритемий. Вряд ли это так, поскольку речь идёт о том, кто первый зафиксировал эту идею письменно. Впрочем, если кто-то ранее него использовал такие системы, то сегодня шифрограммы этих лиц до нас либо не дошли, либо считаются неразгаданными (что очень маловероятно). Так что предположим, что было так – именно Тритемий первым придумал идею биграммного шифра. Впрочем, идея эта настолько проста, что придумать её может кто угодно. Вот я в своё время в школе придумал её именно сам, так что ничего сложного в этом нет.

Иоганн Тритемий – аббат бенедиктинского аббатства, алхимик, каббалист, маг, криптолог, теолог и астролог

Потом биграммные шифры использовались на протяжении всей истории вплоть до начала Второй мировой войны. При этом интерес представляет шифр Плейфера, который был изобретён в 1854 году Чарльзом Уитстоном. Этот шифр вводит правила построения биграмм для шифрования на основе кодового слова, а потому нет необходимости в выдумывании огромного количества замысловатых значков или случайного набора чисел. По ключу, в качестве которого выступает произвольное слово или фраза, шифр восстанавливается однозначно, равно как и правила шифрования и расшифровки. Рекомендую внимательно ознакомиться с описанием этой криптографической системы, а тем, кто умеет программировать, рекомендую также написать программу для шифрования и расшифровки для данной системы по заданному ключу.

Метод атаки

Вдумчивый читатель уже должен был начать догадываться о том, как можно взломать биграммный шифр. Ведь эта схема шифрования не меняет порядок следования букв, она просто превращает два символа в один. Так что количество различных символов, использующихся в шифрограммах, просто стало во много раз больше. Но базовый принцип подсчёта их количества не меняется. Другими словами, *метод частотного анализа* всё так же применим и в случае биграммного шифра. Впрочем, математик вообще скажет, что биграммный шифр ничем от шифра простой подстановки не отличается, и, в общем-то, будет прав.

В математике биграммы обобщаются при помощи так называемых N-грамм. Под N-граммой понимается последовательность из N символов, при этом символ – это вообще отдельная единица переноса смысла или информации. То есть символом может быть звук, фонема, буква, слог, морфема, слово и, в принципе, даже более глобальные единицы передачи смысла – словосочетания, фразы и предложения. Соответственно, N-грамма состоит из N символов. Например, если символом является буква, то примерами 4-грамм являются следующие: РВОА, ОЕГА, ЛЫВО, ПГОЕ и т. д. И, как вы теперь понимаете, биграмма – это просто-напросто 2-грамма.

Соответственно, для N-грамм для различных значений числа N уже составлены частотные словари, которыми можно пользоваться при осуществлении частотного анализа шифрограммы. Для некоторых значений числа N даже построены специальные тематические частотные словари, чтобы можно было проводить криптографический анализ в случае, если известна тема шифрограммы.

Таким образом, для атаки на N-граммный (и, в частности, биграммный) шифр следует воспользоваться следующими правилами:

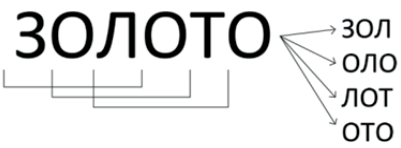
1. Собрать максимально доступный объём зашифрованных материалов – чем больше, тем лучше и проще будет дешифровка.
2. Максимально, если это возможно, осуществить подбор ключевых слов, получив первоначальный набор расшифрованных символов.
3. Применить частотный анализ, строя гипотезы, проверяя их и про-

должая подбирать варианты дешифровки отдельных *N*-грамм.

4. Проверять гипотезы не только на имеющемся материале, но и при помощи контроля частоты межграничных *N*-грамм.

Последний пункт является новым по сравнению с тем, что мы изучили в предыдущей статье относительно шифра простой подстановки (он, кстати, является 1-граммным шифром, как вы теперь должны понимать). Так что давайте внимательно посмотрим на это новое правило.

Рассмотрим слово «ЗОЛОТО». Сколько из него можно сделать 6-грамм? Правильно – одну. А 5-грамм? Две: «ЗОЛОТ» и «ОЛОТО». Соответственно, из этого слова можно сделать три 4-граммы, четыре 3-граммы и пять 2-грамм. Вот все 3-граммы, и как получаются первые три из них:



А вот все 2-граммы и способ получения первых двух из них:

Но теперь давайте подумаем, как это слово будет зашифровано при помощи биграммного шифра? Для него будет использовано три символа, соответствующих 2-граммам «ЗО», «ЛО» и «ТО». Другими словами, при шифровании пропускаются промежуточные *N*-граммы (в случае слова «ЗОЛОТО» – это «ОЛ» и «ОТ»). Но частоты же подсчитаны и для них тоже. Какой из этого можно сделать вывод?

Это всё значит, что в случае подбора вариантов *N*-грамм проверку гипотез можно осуществлять и при помощи сопоставления частот для промежуточных *N*-грамм, которые получаются в процессе. И в этом случае иногда возможно отсечение некоторых гипотез на ранних этапах криптоанализа на основании того, что частоты промежуточных *N*-грамм не совпадают очень сильно. Особенно это возможно в случае автоматизированного анализа с использованием специального программного обеспечения.

Заключение

Таким образом, мы сегодня убили нескольких зайцев. От биграммного шифра мы перешли к рассмотрению *N*-грамм, так что научились взламывать шифры, основанные на замене *N*-грамм. Метод криптоанализа в целом тот же, что используется для шифров простой подстановки, только сложность его возрастает с ростом числа *N*. Можно оценить так, что если для взлома шифрограммы, полученной при помощи шифра простой подстановки, в большинстве случаев достаточно объёма текста в 100 символов, и на это уйдёт час при ручном криптоанализе, то для биграммного шифра минимальный объём текста составляет 1000 символов, а на ручной взлом будет затрачено порядка 10 часов. Для триграммного шифра такие оценки, соответственно, увеличатся до 10 000 символов и 100 часов. Но это, конечно, только в случае ручного взлома. При использовании программного обеспечения всё будет происходить быстрее, но вот требуе-

мые объёмы текста, конечно, не сильно снизятся.

Другими словами, от величины N зависит *порядок* требуемого для успешного криптоанализа объёма шифрограммы и количества времени на криптоанализ. Чем больше значение N, тем больше времени требуется. Так что при довольно больших значениях N взлом шифрограмм может стать делом бесперспективным с точки зрения целесообразности. Тем не менее, при увеличении значения N усилия на шифрование тоже становятся довольно высокими. Этот вопрос оставляю вдумчивому читателю на самостоятельную проработку.

Упражнения

В качестве домашнего задания и самостоятельных упражнений рекомендую выполнить следующее:

1. Прочитайте следующие художественные произведения:

- Стивенсон Н. *Криптономикон.*
- Душкин Р. В. *Криптографические приключения.*

2. Расшифруйте следующее зашифрованное послание:

```
0143 0795 0441 0137 0559 0573 0073 0777 0304 0897 0397 0055
0312 0079 0564 0568 0048 0238 0547 0567 0292 1010 0449 0553
0644 0504 0079 0397 1005 0204 0310 0036 0497 0877 0804 0041
0076 0234 1010 0290 0141 0602 0482 0425 0588 0296 0311 0036
0547 0055 0994 0399 0602 0482 0368 0238 0562 0498 0226 0458
0960 0049 0289 0778 0305 0317 0073 0265 0655 0415 0077 1012
0503 0443 0417 0265 0566 0056 0065 0777 0778 0305 0310 0436
0740 0315 0329 0696 0079 0075 0429 0777 0525 0674 0494 0417
0778 0305 0498 0302 0300 0847 0845 0458 0504 0079 0065 0693
```

В представленной шифрограмме не так много символов, так что частотный анализ провести вряд ли удастся. Тем не менее, внимательное изучение шифрограммы даёт подсказку – если перевести все числа в двоичный 10-значный код, а потом разбить его на пятёрки битов, то частотность полученных пятёрок очень сильно напоминает частотность букв русского языка. Ещё одна подсказка – буквы Е и Ё различаются, пробел не используется.

3. Напишите программу, которая подсчитывает частоты N-грамм для заданного N по предоставленному корпусу текстов. Результаты работы такой программы должны записываться в файл в формате CSV или схожем.

4. Прочитайте описание шифра Плейфера (например, в Википедии). Напишите программу, которая получает на вход текст для шифрования, размер матрицы и ключ, а возвращает шифрограмму, полученную при помощи шифра Плейфера.