СИМЕТРИЧНА КРИПТОГРАФІЯ
КОМП'ЮТЕРНИЙ ПРАКТИКУМ №3

Криптоаналіз афінної біграмної підстановки

Мета роботи

Набуття навичок частотного аналізу на прикладі розкриття моноалфавітної підстановки; опанування прийомами роботи в модулярній арифметиці.

Необхідні теоретичні відомості

1. Опис афінного шифру біграмної заміни

Афінна підстановка біграм являє собою наступне криптографічне перетворення. Відкритий текст $x_1, x_2, x_3, \ldots$, розбивається на біграми, що не перетинаються: (x_1, x_2), (x_3, x_4), …. Нехай m – кількість букв в алфавіті. Занумеруємо букви алфавіту числами від 0 до $m-1$; тоді кожній біграмі (x_{2i-1}, x_{2i}) можна єдиним чином співставити число X_i у границях від 0 до m^2-1:

$$(x_{2i-1}, x_{2i}) \leftrightarrow X_i = x_{2i-1}m + x_{2i}.$$

Наприклад, біграмі $(в, б) = (2,1)$ при $m = 31$ відповідає число $X = 2 \cdot 31 + 1 = 63$.

Біграми шифруються незалежно за таким правилом. Нехай $X_i = (x_{2i-1}, x_{2i})$ – біграма відкритого тексту, а $Y_i = (y_{2i-1}, y_{2i})$ – відповідна біграма шифрованого тексту. Тоді

$$Y_i = (aX_i + b) \bmod m^2,$$

де $0 < a < m^2$ – число, взаємно просте з m, $0 \le b < m^2$ – довільне число у вказаних границях. Таким чином, ключ шифру афінної підстановки складається з пари чисел a і b.

При розшифруванні виконується обернене перетворення:

$$X_i = a^{-1}(Y_i - b) \bmod m^2$$

2. Атака на афінний шифр

Афінна підстановка при атаці, що базується тільки на знанні шифротексту, розкривається за допомогою *частотного аналізу*, що ґрунтується на такому спостереженні: афінний шифр зберігає статистичні властивості мови, пов'язані із частотами біграм.

Нехай з деяких міркувань криптоаналітику стало відомо, що біграма X^* перейшла при шифруванні у біграму Y^*, а біграма X^{**} – у біграму Y^{**}. Тоді для невідомих параметрів ключа a, b можна скласти систему рівнянь:

$$\begin{cases} Y^* \equiv aX^* + b \pmod{m^2} \\ Y^{**} \equiv aX^{**} + b \pmod{m^2} \end{cases}, \quad (1)$$

звідки маємо рівняння для визначення параметру *a*:

$$Y^* - Y^{**} \equiv a(X^* - X^{**}) \pmod{m^2}. \quad (2)$$

Рівняння (2) може мати один або декілька розв'язків. Для кожного можливого значення *a* відповідне значення *b* знаходиться з рівняння

$$b = (Y^* - aX^*) \bmod m^2.$$

Остаточно ключ шифрування знаходиться перебором по знайдених кандидатах із перевіркою на інших біграмах шифротексту.

Звідки ж аналітик може знати про відповідність біграм відкритого та шифротекстів? Він може скористатись тим, що описаний афінний шифр зберігає частоти біграм мови. Якщо позначити через X^* біграму, що найчастіше зустрічається у мові, X^{**} – наступну за нею за частотою і т.д., Y^*, Y^{**} – біграми шифротексту, що також розташовані в порядку спадання частот, то можна очікувати, що біграма X^* під час шифрування переходить у Y^*, а X^{**} – у Y^{**}, звідки й знаходиться ключ шифрування.

Припущення про те, що X^* та X^{**} шифруються відповідно в Y^* та Y^{**}, може виявитися помилковим. Тоді треба підставляти в (1) інші пари *X* та *Y* з числа найбільш імовірних у мові і найчастіших у шифротексті, доки при дешифруванні не вийде змістовний текст.

3. Критерії автоматичного визначення змістовного тексту

При фінальному пошуку з-поміж кандидатів у ключі шифрування, які залишились після аналізу, ми користуємось тим, що правильний ключ при розшифруванні дає змістовний текст, в той час як неправильні ключі будуть давати випадкові тексти (точніше, імовірність, що на неправильному ключі після розшифрування ми одержимо змістовний текст, є вкрай малою). Звісно, коли кандидатів є невелика кількість, то аналітик може просто переглянути їх всі та обрати правильний візуально. Але коли треба перебрати вже сотню варіантів, використання людини в якості розпізнавача мови є негуманним. Тому в програмах криптографічного аналізу потрібно передбачати засоби автоматичного розпізнавання змістовних текстів та відбору їх від текстів випадкових.

В якості засобів розрізняння текстів на природних мовах виступають їх статистичні властивості. Наведемо деякі поширені критерії відбору змістовних текстів.

1) *Критерій заборонених l-грам*. Деякі комбінації символів в природніх мовах не зустрічаються (наприклад, «аь» в українській мові). Отже, поява таких послідовностей у тексті із великою імовірністю свідчить про його випадковість.

Для застосування даного критерію аналітик повинен обрати множину тих *l*-грам, які він вважає забороненими. Існує декілька варіантів використання критеріїв: в першому текст відкидається за наявності в ньому будь-якої із обраних заборонених *l*-грам; в другому текст відкидається, якщо частоти появи кожної *l*-грами не перевищують задане порогове значення; нарешті, в третьому аналітик оцінює сумарну частоту появи в тексті усіх заборонених *l*-грам та відкидає текст, якщо ця сума перевищує заданий поріг.

2) *Критерій частих l-грам* відрізняється від попереднього тим, що досліджуються комбінації символів, які зустрічаються в мові найчастіше (наприклад, «the» для

англійської мови). Відповідно, появи частих *l*-грам в змістовному тексті не повинні бути нижчими за деякий поріг.

В цьому критерії можна відслідковувати як частоту окремих літер, так і біграм, триграм тощо. Але зауважте, що описаний в практикумі афінний шифр зберігає частотний профіль незалежних біграм, на які розбивається відкритий текст. При використанні критерію частих біграм для афінного шифру потрібно досліджувати біграми, взяті на перетині шифрованих пар символів – (x_{2i}, x_{2i+1}) в термінах першого розділу.

3) *Критерій рівномірності l-грам* вивчає кількість *l*-грам, що зустрічаються у тексті. Оскільки природні мови є дуже нерівноімовірними, багато комбінацій символів в них не зустрічається. Тому аналітик може просто обчислити кількість *l*-грам, які не зустрічаються в заданому тексті, і якщо ця кількість менша за деяке порогове значення (яке обирається із урахуванням особливостей мови), то текст відкидається.

4) *Ентропійні критерії* порівнюють значення інтегральних характеристик тексту (таких, як ентропії символів та біграм, індекс відповідності тощо) із еталонними для мови.

5) *Структурний критерій* в простий спосіб визначає, чи має одержаний текст надлишковість: якщо текст добре стискається відомими алгоритмами стиснення даних, то він є змістовним. Випадкові ж тексти зазвичай не стискаються взагалі, оскільки не мають надлишковості, притаманної природним мовам.

Існує і безліч інших критеріїв, що виявляють ті чи інші властивості природної мови; також можливе застосування довільних комбінацій цих методів. Якість відбору залежить від налаштування параметрів (порогових та еталонних значень).

4. Додаткові відомості з теорії чисел

Рівність (2) є так званим *лінійним порівнянням*; розв'язки лінійних порівнянь знаходяться за такою процедурою.

Нехай $ax \equiv b \pmod n$ і треба встановити значення *x* за відомими *a* та *b*. Маємо такі випадки:

1) $\gcd(a,n) = 1$. В цьому випадку порівняння має один розв'язок: $x \equiv a^{-1}b \pmod n$.

2) $\gcd(a,n) = d > 1$. Маємо дві можливості:

   2.1) Якщо *b* не ділиться на *d*, то порівняння не має розв'язків.

   2.2) Якщо *b* ділиться на *d*, то порівняння має рівно *d* розв'язків x_0, $x_0 + n_1$, $x_0 + 2n_1 \ldots$, $x_0 + (d-1)n_1$, де $a = a_1 d$, $b = b_1 d$, $n = n_1 d$ і x_0 є єдиним розв'язком порівняння $a_1 x \equiv b_1 \pmod{n_1}$: $x_0 = b_1 \cdot a_1^{-1} \bmod n_1$.

Для обчислення обернених елементів за даним модулем пропонується використовувати розширений алгоритм Евкліда.

Нагадаємо, що алгоритм Евкліда обчислює найбільший спільний дільник двох чисел $d = \gcd(a,b)$ таким чином. Задаємо $r_0 = a$, $r_1 = b$ та обчислюємо послідовність (r_i) для $i \geq 2$ шляхом ділення з остачею:

$$
\begin{aligned}
&r_0 = r_1 q_1 + r_2,\\
&r_1 = r_2 q_2 + r_3,\\
&\ldots\\
&r_{s-2} = r_{s-1} q_{s-1} + r_s;\\
&r_{s-1} = r_s q_s.
\end{aligned}
$$

Якщо на відповідному кроці виявилось, що $r_{s+1} = 0$, то $d = r_s$.

Розширений алгоритм Евкліда обчислює дві додаткові послідовності (u_i) та (v_i) такі, що на кожному кроці виконується рівність $r_i = u_i a + v_i b$; зокрема, для найбільшого спільного дільника матимемо $d = r_s = u_s a + v_s b$. Ці послідовності також можна обчислити рекурентно за допомогою часток q_i:

$$u_0 = 1,\ u_1 = 0,\ u_{i+1} = u_{i-1} - q_i u_i;$$
$$v_0 = 0,\ v_1 = 1,\ v_{i+1} = v_{i-1} - q_i v_i.$$

Звідси обернений елемент до числа *a* за модулем *n* знаходиться таким чином: оскільки *a* обертається лише за умови $\gcd(a,n) = 1$, то за розширеним алгоритмом Евкліда знаходяться такі числа *u* та *v*, що $au + nv = 1$. Звідси $au \equiv 1 \pmod n$ та $u \equiv a^{-1} \pmod n$.

Порядок виконання роботи

0. Уважно прочитати методичні вказівки до виконання комп'ютерного практикуму.

1. Реалізувати підпрограми із необхідними математичними операціями: обчисленням оберненого елементу за модулем із використанням розширеного алгоритму Евкліда, розв'язуванням лінійних порівнянь. При розв'язуванні порівнянь потрібно коректно обробляти випадок із декількома розв'язками, повертаючи їх усі.

2. За допомогою програми обчислення частот біграм, яка написана в ході виконання комп'ютерного практикуму №1, знайти 5 найчастіших біграм запропонованого шифротексту (за варіантом).

3. Перебрати можливі варіанти співставлення частих біграм мови та частих біграм шифротексту (розглядаючи пари біграм із п'яти найчастіших). Для кожного співставлення знайти можливі кандидати на ключ (a,b) шляхом розв'язання системи (1).

4. Для кожного кандидата на ключ дешифрувати шифротекст. Якщо шифротекст не є змістовним текстом російською мовою, відкинути цього кандидата.

5. Повторювати дії 3-4 доти, доки дешифрований текст не буде змістовним. У разі необхідності змінити кодування алфавіту (див. методичні вказівки).

Методичні вказівки

Студентам надається текст, що є результатом шифрування за допомогою афінної підстановки біграм відкритого тексту, написаного російською мовою без пробілів, знаків пунктуації та великих літер. Буква «ё» заміщена буквою «е», а «ъ» – буквою «ь» (або навпаки). Таким чином, алфавіт відкритого тексту складається з 31 букви, що занумеровані в алфавітному порядку: *а* = 0, *б* = 1, …, *я* = 30.

Зауваження: у деяких варіантах літера «ь» стоїть у алфавіті після літери «ы» (що відповідає кодуванню *ы* = 26, *ь* = 27), а в деяких навпаки, «ь» ототожнюється із літерою «ъ» та стоїть перед літерою «ы» (що відповідає кодуванню *ь* = 26, *ы* = 27). Під час роботи вам необхідно встановити, яким саме способом закодовано алфавіт, для коректного дешифрування тексту.

П'ятьма найчастішими біграмами російської мови (в порядку спадання частот) є біграми «ст», «но», «то», «на», «ен». Перевірте ці відомості за допомогою програми підрахунку частот біграм з комп'ютерного практикуму №1.

Під час дешифрування виникне потреба відрізнити змістовний текст російською мовою від тексту-шуму, що виникає при неправильному дешифруванні. Вважаючи на доволі велику кількість можливих варіантів ключів, для цієї задачі необхідно побудувати автоматичний розпізнавач російської мови. Створення та принцип роботи такого

розпізнавача залишається на ваш розсуд; традиційно використовують такі критерії змістовного тексту:

- перевірку частот частих літер («о», «а», «е», частоти можуть розглядатись окремо або в сукупності);
- перевірку частот рідкісних літер («ф», «щ», «ь», частоти також можуть розглядатись окремо або в сукупності);
- перевірку частот біграм, підраховану для біграм «на перетині» (у вищенаведених позначеннях – біграм виду (x_{2i}, x_{2i+1}));
- перевірку частот триграм та довільних l-грам.

Зашифровані файли із варіантами завдань містяться в папці “variants”. Зашифровані файли, що містяться у папці “for_test”, є більш простими для аналізу, ніж основні варіанти, їх можна використовувати для тестування або налаштування ваших програм.

Оформлення звіту

Звіт до комп’ютерного практикуму оформлюється згідно зі стандартними правилами оформлення наукових робіт, за такими винятками:

- дозволяється використовувати шрифт Times New Roman 12pt та одинарний інтервал між рядками;
- для оформлення фрагментів текстів програм дозволяється використовувати шрифт Courier New 10pt та друкувати тексти в дві колонки;
- дозволяється не починати нові розділи з окремої сторінки.

До звіту можна не включати анотацію, перелік термінів та позначень та перелік використаних джерел. Також не обов’язково оформлювати зміст.

Звіт має містити:

- мету комп’ютерного практикуму;
- постановку задачі та варіант завдання;
- хід роботи, опис труднощів, що виникали, та шляхів їх розв’язання;
- знайдені п’ять найчастіших біграм шифротексту;
- опис роботи запропонованого вами автоматичного розпізнавача російської мови (із обґрунтуванням коректності);
- шифрований та відповідний розшифрований тексти (відповідно до варіанту завдання), знайдене значення ключа;
- висновки.

Тексти всіх програм здаються викладачеві в електронному вигляді для перевірки на плагіат. До захисту комп’ютерного практикуму допускаються тільки ті студенти, які оформили звіт та пройшли перевірку програмного коду.

Контрольні запитання

1) Що таке шифр афінної підстановки взагалі та афінної підстановки біграм зокрема? Опишіть процес зашифрування та розшифрування.

2) Що таке шифри моно- та поліалфавітної підстановки? До якого класу відносяться шифри афінної підстановки?

3) Скільки ключів може мати шифр афінної підстановки біграм на введеному у роботі алфавіті?

4) Опишіть роботу розширеного алгоритму Евкліда та обґрунтуйте його коректність.

5) Чому для існування оберненого елементу необхідно, щоб елемент був взаємно простим із модулем?

6) Яким чином розв’язуються лінійні порівняння?

7) Які критерії змістовного тексту ви знаєте?

Оцінювання практикуму

За виконання комп’ютерного практикуму студент може одержати до 7 рейтингових балів; зокрема, оцінюються такі позиції:

- реалізація програм – до трьох балів (в залежності від правильності та швидкодії);
- теоретичний захист роботи – до трьох балів;
- несвоєчасне виконання роботи – (-1) бал за кожні два тижні пропуску дедлайну.

Програмний код, створений під час виконання комп’ютерного практикуму, перевіряється на наявність неправомірних запозичень (плагіату) за допомогою сервісу *Stanford MOSS Antiplagiarism*. У разі виявлення в програмному коді неправомірних запозичень реалізація програм оцінюється у 0 балів, а за виконання практикуму студент одержує штраф (–10) балів.

Студенти допускаються до теоретичного захисту тільки за умови оформленого звіту з виконання практикуму та проходження перевірки програмного коду.