Genetic Algorithms for Frequency Analysis in Breaking Substitution Ciphers Problem

I. Khalina, A. Usatov

***Abstract*—** In cryptoanalysis, frequency analysis is a common method for breaking substitution ciphers. The approach is based on the fact that natural languages possess statistical regularities. In particular, for each natural language there is a distribution of letter occurrence frequencies which persists for almost all language samples. The idea behind decryption ciphertexts encrypted by substitution ciphers using frequency analysis is to associate frequencies with which letters occur in ciphertext with corresponding ones in plaintext. The purpose of the work is to develop an algorithm that builds the most accurate decryption map based on frequency analysis of N-gram language model.

For solving the problem, the approach of genetic algorithms was chosen. To calculate N-gram occurrence frequencies in English language the text corpora of more than 200 000 words was processed. A random decryption map was generated. Discrepancies between generated decryption map and estimated decryption map was minimized iteratively.

Lastly, the result of decoding ciphertext with built decryption map was compared to original plaintext to demonstrate its accuracy.

***Keywords*—** Cryptoanalysis, frequency analysis, genetic algorithms, N-gram language model.

I. Introduction

Substitution ciphers key can be defined as a bijection where each element of alphabet is independently associated with another. Thus, the number of possible keys is equal to a factorial of the cardinality of the alphabet.

However, the use of statistical and linguistic methods in frequency analysis allows obtaining additional information about the key, which can be used to solve the key search problem.

The analysis of the cryptosystem and the search for the encryption transformation key can be divided into two stages: the construction of an N-gram language model based on the results of frequency analysis of large text corpora and the development of a genetic algorithm using the constructed model.

II. N-gram Language Model

Let $x_1x_2 \dots x_m$ be a character sequence consisting of m elements. The idea of the N-gram model is to approximate the conditional probability of occurrence of a certain character of the alphabet at the k-th position $P(x_k|x_1 \dots x_{k-1})$ by calculating the conditional probability using a sequence of preceding characters of a fixed length $N-1$.

Let the symbols of the natural language alphabet be the states of the event of the appearance of a certain character at the k-th position, then assuming the appearance of a sequence of consecutive characters in the text as a Markov chain and generalizing the Markov property for a sequence of $N-1$ events [1], we obtain an approximation

$$P(x_k|x_1 \dots x_{k-1}) \approx P(x_k|x_{k-N+1} \dots x_{k-1}) \quad (1)$$

Thus, the formula for calculating the probability of the appearance of the entire sequence of m characters in the text for $N > 2$ will take the form

$$P(x_1x_2 \dots x_m) = P(x_1) \dots P(x_{N-1}|x_1 \dots x_{N-2}) \prod_{k=N}^{m} P(x_k|x_{k-N+1} \dots x_{k-1}) \quad (2)$$

I. Khalina, Belarusian State University, Minsk, Belarus (e-mail: fpm.halina@bsu.by).
A. Usatov, Belarusian State University, Minsk, Belarus (e-mail: ausatov@icloud.com).

III. Genetic Algorithm

Genetic algorithms – is a class of evolutionary algorithms designed to solve optimization and modeling problems using analogues of the mechanisms of genetic inheritance and natural selection.

At the same time, genetic algorithms have the following properties that characterize their stability:

• operations on populations – search for a solution from a certain population;

• using a minimum of information about the task – only the objective function is used, and not its derivative or other information about the task;

• randomization – probabilistic rather than deterministic selection rules are applied.

Fig. 1 illustrates general stages of genetic algorithms.

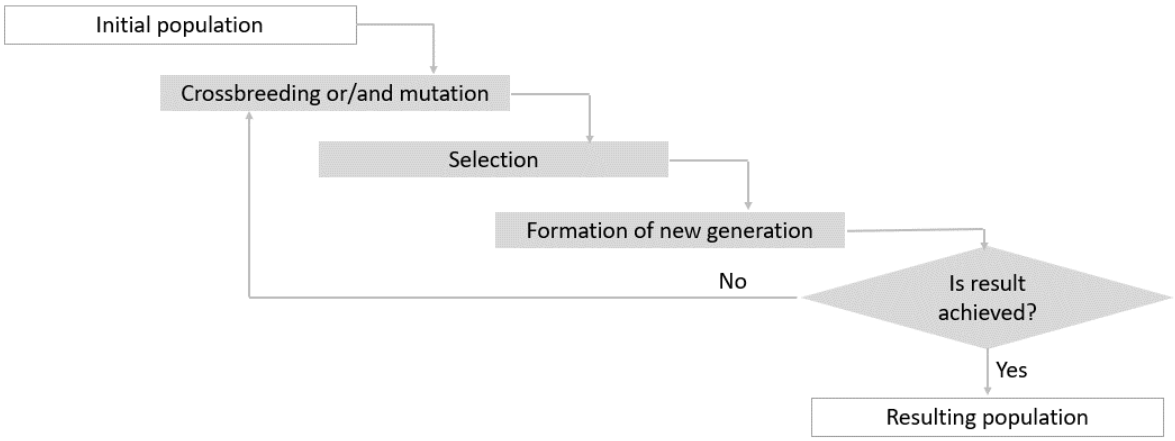

Fig. 1 General scheme of genetic algorithms

At the initialization stage, a population of individuals is randomly generated – a vector containing n possible solutions to the problem [2].

The assessment of the fitness of each of the individuals in the population is made by calculating the value of the fitness function.

From the population, k individuals-parents are selected, which will participate in the process of reproduction. The choice is made according to the principle of natural selection, that is, the probability of breeding is higher in individuals with better fitness [3].

To produce offspring among the selected parents, pairs are selected to which the crossover operator is applied. The result of applying the crossing-over operation to a pair of parent individuals is two descendant individuals. The mutation operator is applied to the resulting descendant individuals [2].

Among the descendant individuals, using the selection operator, n individuals are selected, which will form a new population.

The iterative process of the genetic algorithm continues until the termination criterion is reached.

The population obtained at the end of the iterative process is called the resulting population. Among the resulting population, an individual with the best value of the fitness function is selected. This individual is the best solution to the problem.

IV. Key Search Algorithm

The purpose of the developed algorithm is to approximate the encryption transformation key based on the ciphertext. Input data format: ciphertext encrypted using the substitution cipher cryptosystem. Output data format: approximate encryption transformation key.

To construct N-gram language model the novel by Fyodor Mikhailovich Dostoevsky "Crime and Punishment" was chosen as the text corpora for analysis. Its English version includes more than 205 000 words, which makes it large enough to obtain a high frequency distribution accuracy in the analysis.

At the preprocessing stage, the following text transformations were carried out:

1) converting all text characters to lower case;

2) removal from the text of all characters other than the characters of the English alphabet and a space character, as well as all punctuation marks;

3) removal of stop words that may affect the resulting distribution, such as Russian names, surnames and patronymics;

4) text tokenization.

The result of preprocessing is a filtered list of words in the order of their appearance in the text of the novel, suitable for further analysis.

To calculate the maximum likelihood of N-grams in the process of text analysis, the frequency of occurrence of each of the N-grams is calculated, which is subsequently normalized by dividing by the sum of the frequencies of all N-grams that have a common beginning [1]. Therefore, the maximum likelihood of word can be calculated using formula

$$P(x_1 x_2 \dots x_m) = \frac{\mathrm{C}(x_1)}{\mathrm{C}(x)} \dots \frac{\mathrm{C}(x_1 \dots x_{N-2} x_{N-1})}{\mathrm{C}(x_1 \dots x_{N-2} x)} \prod_{k=N}^{m} \frac{\mathrm{C}(x_{k-N+1} \dots x_{k-1} x_k)}{\mathrm{C}(x_{k-N+1} \dots x_{k-1} x)} \quad (3)$$

where C is the number of occurrences of sequence given in parentheses in text.

Let $w_1 w_2 \dots w_n$ be a sequence of natural language words, or a message. Assuming that the appearance of each individual word is an independent event, the probability of the appearance of the entire message can be calculated from the corollary of the probability multiplication theorem for independent events

$$P(w_1 w_2 \dots w_n) = \prod_{i=1}^{n} P(w_i) \quad (4)$$

To calculate the probability of a message appearing according to formulas (3)-(4), it is necessary to multiply multiple floating-point numbers, which can lead to data loss due to overflow and a significant decrease in the accuracy of calculations. To avoid this, instead of calculating the probability, the logarithm of the probability is calculated, which will allow replacing multiplication with addition [1].

The first step in building a genetic algorithm is to determine the object that represents the individual. The purpose of applying a genetic algorithm for this problem is to find the encryption transformation key. Therefore, an individual, that is, a solution, will be a real-coded vector containing a sequence of English alphabet characters [4]. In addition, due to the bijectivity of substitutions, the vector must include all possible characters of the English alphabet only once [5].

The second step is to determine the type of fitness function that will associate each individual with a numerical value, on the basis of which the selection will be made.

To determine the fitness of an individual, that is, a key, it is possible to estimate the probability of a message decrypted on a given key appearing in the plain text.

When building the N-gram model, the likelihood function was derived, which determines the probability of a message appearing.

Thus, the fitness function defined for this algorithm associates each individual with the probability of the appearance of a message decrypted on a given key, and the problem of finding a key is reduced to the problem of maximizing the likelihood function.

A combination of elitist selection and steady state selection is chosen as selection operators.

To fulfill the condition of bijective substitution, offspring must be produced from only one parent, so the crossing-over step can be neglected, obtaining offspring only by using the mutation operator.

Due to the uncertainty of the exact maximum of the likelihood function due to the peculiarities of natural languages, as a criterion for the completion of the iterative process, we can define the absence of progress (the absence of the appearance of an individual whose fitness function value is greater than that of all current individuals) during a certain number of iterations.

Taking into account all the above features of the algorithm, the process of its work can be described as follows:

1) generation of the initial population of individuals representing the encryption transformation key;

2) calculation for each individual in the population of the value of the likelihood function for the message decrypted on the given key;

3) selection using the strategy of elite selection of individuals for reproduction;

4) mutation of selected individuals by swapping two randomly selected genes;

5) combining individuals of the current generation and individuals-descendants;

6) formation of a new population by elitist selection among all individuals;

7) repeating steps 2-6 until the completion criterion of the iterative process is met;

8) selection from the resulting population of an individual with the best fitness.

V. Evaluation of the Quality of the Algorithm

The performance of the algorithm was evaluated by calculating the Hemming distance between the real encryption transformation key and the approximated by the algorithm to determine their difference.

Fig. 2 shows the results of one of the test runs of the algorithm: the original plaintext and the plaintext obtained by decrypting the ciphertext with the key approximated by the algorithm, actual key and generated key and Hemming distance between them.

```
Actual key:        m y o e n a v x u h k c t z f i g j r l d q s p b w
Generated key:     m y o e n a v x u h k c t z f i g j r l d q s w b p
Hemming distance:  2

Original message:

He smiled understandingly much more than understandingly.
It was one of those rare smiles with a quality of eternal reassurance in it,
that you may come across four or five times in life.
It faced or seemed to face the whole eternal world for an instant,
and then concentrated on you with an irresistible prejudice in your favor.
It understood you just as far as you wanted to be understood,
believed in you as you would like to believe in yourself,
and assured you that it had precisely the impression of you that,
at your best, you hoped to convey.


Decoded message:

he smiled understandingly much more than understandingly
it was one of those rare smiles with a quality of eternal reassurance in it
that you may come across four or five times in life
it faced or seemed to face the whole eternal world for an instant
and then concentrated on you with an irresistible prejudice in your favor
it understood you just as far as you wanted to be understood
believed in you as you would like to believe in yourself
and assured you that it had precisely the impression of you that
at your bestyou hoped to convey
```

Fig. 2 Test run results

Thus, the Hamming distance for this test run was two positions. It is to be noted that letters which positions in the substitution were determined incorrectly are either absent in the original message or are extremely rare.

Fig. 3 shows graphs showing the fitness progress of the best individual in the population and the progress of average fitness across the population.

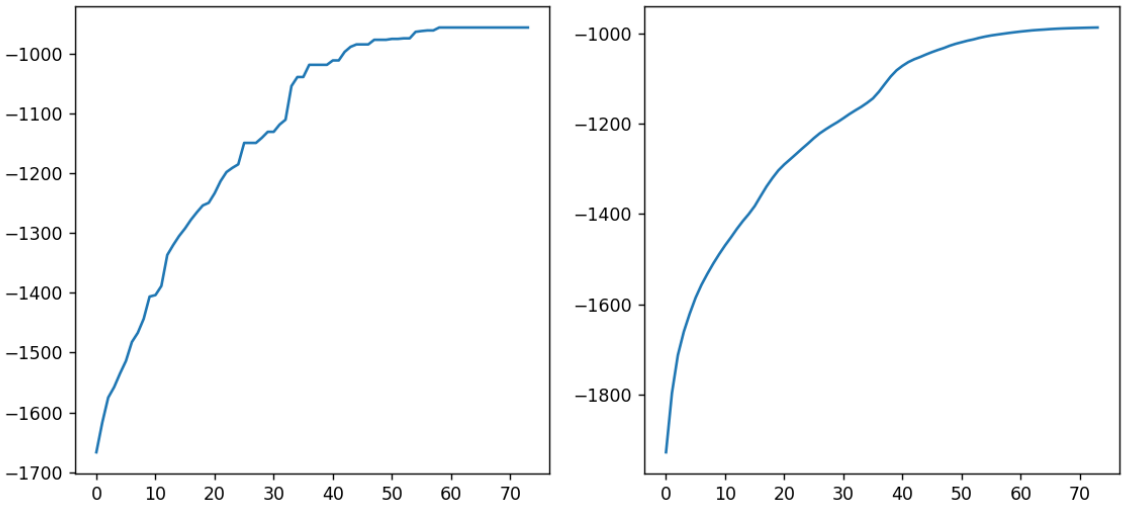

Fig. 3 Best fitness per iteration and average fitness per iteration

A metric for evaluating the performance of the algorithm without using information about the actual encryption transformation key was calculated as follows:

1) probability of each word in the decrypted message was normalized by calculating geometric mean of N-gram probabilities;

2) probability of the decrypted message was normalized by calculating geometric mean of normalized probabilities of words;

3) logarithm of normalized probability of the message was used to replace multiplication with addition.

Due to normalization this metric allows to compare the performance of the algorithm for messages of different lengths.

Fig. 4 shows the evaluation of the best individual at each iteration at one of test runs.

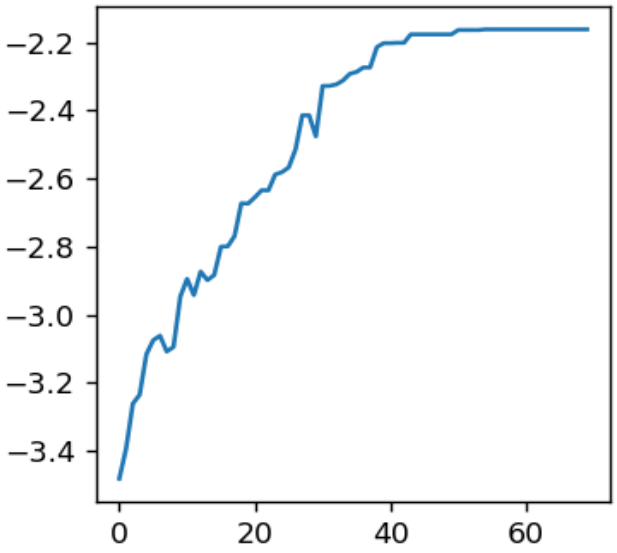

Fig. 4 Best individual evaluation per iteration

Fig. 5 shows the evaluation of the results of test runs with messages of different lengths.

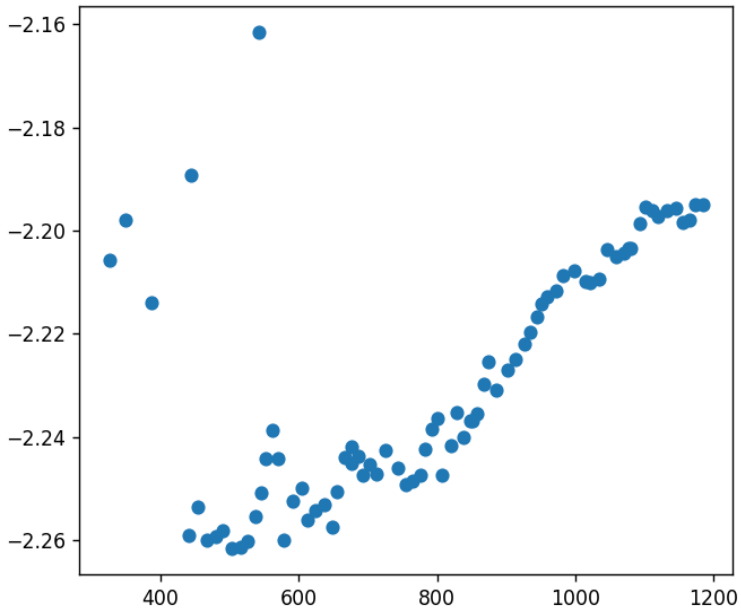

Fig. 5 Evaluation of solution for messages of different lengths

It should also be noted that the developed algorithm does not converge for small messages (less than three hundred characters), since the distribution of character frequencies for texts of this length can differ significantly from the natural language frequency distribution. Also, due to the stylistic features of natural languages, the analysis of an artistic text is effective for solving the problem of finding a key only if the style of the encrypted message is artistic or journalistic.

REFERENCES

[1] D. Jurafsky, J. H. Martin, "*Speech and Language Processing"*, 3d ed. draft, unpublished, pp. 32–39.
[2] M. Mitchell, "*An Introduction to Genetic Algorithms"*, 1st ed., Cambridge: MIT Press, 1998, pp. 7-12.
[3] T. Alam, Sh. Qamar, A. Dixit, M. Benaida, "*Genetic Algorithm: Reviews, Implementations, and Applications"*, International Journal of Engineering Pedagogy (iJEP), 2020
[4] U. Bodenhofer, "*Genetic Algorithms: Theory and Applications"*, 3d ed., unpublished, pp. 59-63.
[5] T. Jacobsen, "*A Fast Method for the Cryptanalysis of Substitution Ciphers"*, Cryptologia, 1995