Uppsala University

Cryptology

Group 12

Report for Lab B-C

Authors:
Hamit Efe Çınar

February 8, 2024

1 Part B

1.1 How to Run the Source Code

The Python packages scipy and collections should be installed to run the source code. In the directory, VigenereLab (which can be unzipped from VigenereLab.zip), there are two directories named "ciphertexts (short key)", "ciphertexts (long key)"; and there are two python files named "b1.py", "b2.py". The file "b1.py" uses the crypto text files in the "ciphertexts (short key)" directory as it refers to the solution for the first part of the Lab B. The file "b2.py" uses the crypto text files in the "ciphertexts (long key)" directory as it refers to the solution for the second part of the Lab B.

1.2 The Techniques Used

To solve the cipher texts, Friedman Test is used for both cases to determine the key length of each cipher text. Also, Frequency Analysis is used to find the each index of the key. For this purpose, Chi-Square test is used to determine the deviation from the expected distribution of letter frequencies for Swedish. After the key is found, a standard Vigenere decryption function is used to determine the plain texts.

1.3 The Decryption Process in Detail

For the first part of the assignment B, where the plain texts for the cipher texts with short keys are requested, a several operations are done to solve the problem. First, the Friedman Test is applied to each cipher text to determine the key length of every cipher text. It is done by iterating over the range of possible key lengths, and then calculating the index of coincidence value of the segments created with that key length value. The key length that gives the closest index of coincidence result to the Swedish's is chosen to be the key length of that cipher. The range for iteration is determined to be 1 to 16, as it is given in the assignment paper. The index of coincidence value for Swedish is denoted as 0.0681 in this website.

After that, the cipher texts are split into segments by their corresponding key lengths to apply frequency analysis to each region that is affected by a different index of key separately. This list of segment lists is named "segment

matrix" in the source code. Then, the "calculate shifts" function calculates the shifts needed to decrypt each segment of the Vigenère cipher. It uses the chi-square test to compare observed and expected letter frequencies. The expected letter frequencies for the Swedish Language is taken from this website (Frequency Table), and is normalized in the code to exploit the stats library's chi-square function. The shifting letter that gives the least chi-square value is selected as the corresponding index of its key, and the key is formed afterwards. As some of the lengths of the cipher texts are not long enough compared with its key, the "reconstruct key" function finds some of the keys with slight mistakes; but as the errors are fairly small, it could be manually adjusted to a meaningful Swedish expression. After that, the keys are used to decrypt cipher texts to their corresponding plain texts. Every key and plain text is found without an error via the source code. The operations for this part of the assignment lasts approximately 2 seconds.

For the second part of the assignment, similar operations were done to find the plain texts of given cipher texts. However, as the key is same for each cipher text, the cipher texts were concatenated after the key length is found in order to make the observed distribution of letter frequencies to the expected Swedish letter frequencies. For this concatenation, the "text sharpener" function is used in the source code, which takes the modulo of the cipher texts by the key length, and disregard the remainder tail of the corresponding texts to preserve the indexes affected by the key. The range of the key length is assumed to be 100 to 200 in the source code, and is found to be 123. The remaining operations were the same as the first part. As the key is found, the cipher texts were successfully decrypted to their plain text versions. The operations for this part of the assignment lasts approximately 1.5 seconds.

2 Part C

2.1 Question 1

Breaking a modified Vigenère cipher with 29 characters instead of 26 can be slightly more difficult than the original version with 26 characters since it increases the sample space, but it still relies on the same principles of frequency analysis and guessing the key. With more characters, there are

more possibilities for the key, which could make it marginally harder to break without knowing the key or key length. However, the fundamental approach to breaking it would remain similar. The main factors affecting the difficulty of breaking the cipher are the length of the key and the length of the cipher text. The key is again found with slight mistakes, but as the meaning was clear from the general picture, those 3-5 mistaken indexes were manually corrected in the source code. It takes 2

2.2 Question 2

The length of the key in the Vigenère cipher directly impacts its security. A longer key makes the encryption stronger and harder to break. With a longer key, patterns in the ciphertext become less evident, making frequency analysis and other attacks more challenging. Additionally, the key should be chosen randomly to make it unpredictable while solving.

2.3 Question 3

To break the Vigenère cipher, I would do Kasiski test, which looks for repeating patterns in the ciphertext, and index of coincidence analysis can still be used to identify potential key lengths and patterns in the ciphertext. Without knowing the index of coincidence of the language, it would not be sufficient just to do the index of coincidence analysis. However, considering the possible key lengths coming from the Kasiski text in the index of coincidence analysis, one may derive the language among the index of coincidence values of the world's major languages.

2.4 Question 4

2.4.1 Reverse Key

This proposed cipher defines a cryptosystem since it involves an encryption algorithm, a decryption algorithm, and a key generation method. The security of this cipher depends on the security of the one-time pad, which is theoretically unbreakable if used correctly. However, the security of the one-time pad relies on using truly random keys that are at least as long as the plaintext. In this proposed cipher, the key is the reverse of the plaintext, which may not meet the criteria of being truly random. Knowing that the

cipher text is encrypted with this algorithm, an attacker can try to brute force the possibilities by following this approach: We know that the i'th index is encrypted with N-i'th index of the text where N is the text length, and vice-versa. If the sum of key values of these indexes are 2k, without loss of generality we can say that there are k + (13 + k) = 2k + 13 pairs for these two indexes. So, the cost of the brute force attack is $(2k + 13)^{N/2}$. As 2k + 13 is a constant, the asymptotic running time is exponential. Though as can be observed, even though it might be infeasible to find the meaningful plain text for long texts, it is fairly easy to implement a brute-force decryption algorithm and it is easy to solve short texts in a reasonable time. So it can be concluded that, this algorithm is more secure than Vigenere Cipher for long texts.

2.4.2 Vigenere Cipher with Substraction

This proposed cipher defines a cryptosystem as it still involves an encryption algorithm, a decryption algorithm, and a key generation method. Since the only difference is the direction that the key is shifting; the computational complexity, thus the secureness, of this cipher is the same as the original Vigenere Cipher. Thus, the attacking approach is also the same as we did in the Vigenere Lab B.

2.4.3 Modulo Applied Plain Text

This proposed cipher defines a cryptosystem as it still involves an encryption algorithm, a decryption algorithm, and a key generation method. The computational complexity and the attacking approach is still the same with Vigenere Cipher. The only thing that is needed to be changed from the Vigenere Lab B is that substituting the frequency ratio of letter with their difference modulo in the frequency table, and of course we need to substitute with the corresponding letter for each index of the plain text at the end. Thus, the secureness is the same with the Vigenere Cipher.

2.4.4 Key = Key + Plain Text

This proposed cipher still defines a cryptosystem as it involves an encryption algorithm, a decryption algorithm, and a key generation method. By appending the plaintext to the key stream, the key becomes longer and potentially

less predictable. This could enhance security compared to the Vigenère cipher, especially if the plaintext has high entropy or randomness. However, if the plaintext has patterns or repetitions, they could potentially weaken the security of the cipher. The key space for this cipher would depend on the length of the plaintext concatenated with the initial key. If both the key and the plaintext are sufficiently long and random, the key space would be large, making a brute force attack computationally infeasible. The cost of a brute force attack would still be exponential in terms of the combined length of the key and the plaintext. If the key is relatively small against the text, traditional cryptanalysis techniques such as frequency analysis may still be applicable. Also, as mentioned above, the repetitions that are in the plain text can be exploited by the attacker.

2.4.5 Key = Key + Cipher Text

This proposed cipher still defines a cryptosystem as it involves an encryption algorithm, a decryption algorithm, and a key generation method. It is less secure than Vigenere Cipher since brute forcing a small portion of the cipher text, which is affected by the initial key, is enough to decipher the whole text. It is because of the fact that it takes polynomial time to find where the part that is affected by the initial key ends in the cipher text by decrypting with the beginning of the cipher text. Since the brute force volume is smaller, the running time of an attack is smaller. Thus, it is less secure than the Vigenere Cipher.

2.5 Evaluation for the Lab

Though it was easy to implement the Friedman test, I found it challenging to implement the chi-square test using the expected and observed frequency distributions. Overall, it was quiet fun to work on this assignment.