Solving a 750-Letter General Bigram Substitution Challenge

Elonka Dunin
Codebreaking-guide.com
elonka@gmail.com

Jarl Van Eycke
Independent Researcher
jarlve@yahoo.com

Louie Helm
RockstarResearch.com
louiehelm@protonmail.ch

Klaus Schmeh
Codebreaking-guide.com
klaus@schmeh.org

Abstract

The general bigram substitution cipher is an encryption method originating in the Renaissance. It operates using a substitution table that maps each possible letter pair (bigram) to a unique replacement. While conceptually straightforward, this cipher is notably challenging to break, particularly when dealing with short ciphertexts. To inspire further research, one of the authors initiated a bigram substitution challenge featuring a 750-character ciphertext. In this paper, we present the solution to that challenge, achieved by two other authors using a hill climbing algorithm combined with a scoring function based on 8-gram (eight-letter sequence) frequencies. Since no prior 8-gram frequency statistics existed for the English language, one of the authors developed a comprehensive dataset by analyzing 2 terabytes of text, including 5.8 million books and the entire content of Wikipedia. This achievement, to our knowledge, marks the shortest bigram substitution ciphertext ever successfully decrypted. Furthermore, we propose a new challenge based on a 600-character ciphertext and invite readers to tackle it, setting the stage for future advancements in this field.

1 Introduction

A bigram, or digraph, is a pair of letters, such as PC, QE, IE, or WW. In the commonly used 26-letter Latin alphabet, there are 676 possible bigrams, calculated as 26×26. The general bigram substitution cipher is an encryption technique that replaces each bigram with another bigram, a symbol, or a sequence of characters, following a predefined substitution table. This table can be constructed using a password or another mnemonic method. However, for the purposes of this paper, we assume that the substitution table is always generated randomly.

The earliest known general bigram substitution cipher was introduced by Giovanni Battista Porta in his 1563 book “De Furtivus Literarum Notis” (Kahn, 1996). Porta's system utilized a 20-letter alphabet, resulting in a substitution table with 400 entries (Figure 1). To implement this, Porta employed 400 distinct glyphs in his substitution table. In contrast, the bigram substitutions discussed in this paper involve replacing letter pairs directly with other letter pairs, adhering to a more modern approach.

2 The Playfair cipher

A variety of specialized versions of the general bigram substitution can be designed by replacing an exhaustive substitution table with a more concise set of rules. A well-known method of this type is the Playfair cipher, which operates on a 25-letter alphabet (Dunin and Schmeh, 2023). This cipher employs a simple yet effective set of four substitution rules applied to a 5×5 letter matrix, making it significantly more practical than a general bigram substitution system. The matrix, which serves as the cipher's key, can either be generated from a keyword or selected at random—the latter offering greater security.

As with all substitution ciphers, cryptanalysis of the Playfair cipher becomes more difficult if there is less ciphertext to analyze. The shortest

Proceedings of the 8th International Conference on Historical Cryptology, pages 133-140
Poznań, Poland, 16-18 June, 2025

random-matrix Playfair ciphertext that has ever been broken consists of 26 letters (Dunin et al. 2021).

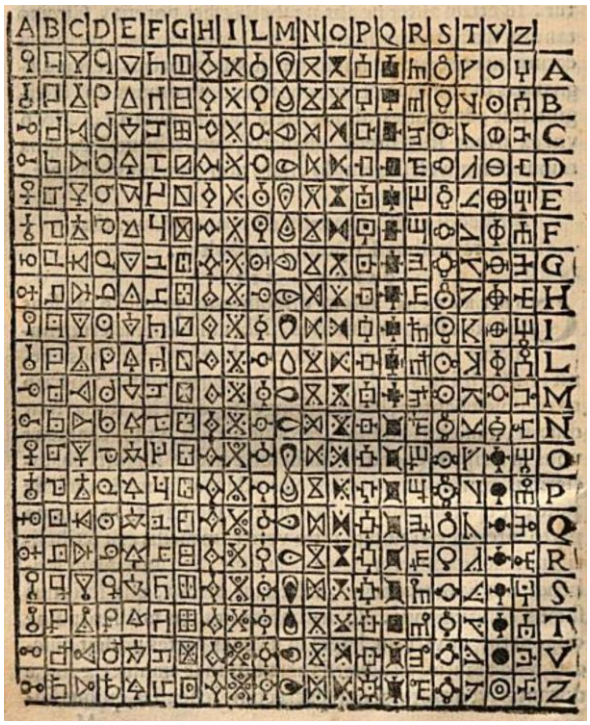

Figure 1. The earliest known general bigram substitution, published in the 1563 book “De Furtivus Literarum Notis” by Giovanni Battista Porta.

The general bigram substitution has never attained the widespread popularity of the Playfair cipher, likely due to its greater complexity. However, it is evident that the general bigram substitution offers a higher level of security compared to the Playfair cipher, though the exact extent of this advantage remains uncertain. Until recently, detailed information regarding the cryptanalysis of the general bigram substitution was notably lacking.

3 The first three challenges

To inspire research on this question, in 2017 one of the authors (Klaus) published two challenges (Schmeh, 2017): a 5000-letter message and a 2500-letter message, both encrypted with a general bigram substitution. Norbert Bierman solved these challenges within a few days. Other interesting comments came from Thomas Bosbach, Thomas Ernst, George Lasry, and Armin Krauß. The method used to break the two ciphertexts was hill climbing (including simulated annealing) with a scoring function based on 4-gram and 5-gram frequencies. To our knowledge, the 2500-letter cryptogram became the shortest one of its kind ever solved.

Building on this success, Klaus introduced a third general bigram challenge ciphertext, comprising 1,346 letters (Schmeh, 2019a). Norbert Biermann deciphered it within a few weeks (Schmeh, 2019b), setting another new world record. Once again, the breakthrough was achieved using hill climbing. This time, the scoring function was based on 6-gram frequencies.

4 The 1000-letter challenge

Next, Klaus created a fourth challenge. This time, he took a plaintext consisting of exactly 1000 letters and encrypted it with a general bigram substitution. In October 2019, he published the resulting cryptogram online (Schmeh, 2019c). Again, the challenge was broken within days, which meant that a new world record was set (Schmeh, 2019d). This time, the solution came from Jarl and Louie, who are among the authors of this paper. Richard Bean, George Lasry, Dave Oranchak, and Christoph Tenzer provided helpful comments.

Jarl is the developer of the software package AZdecrypt (Van Eycke, 2016). Initially conceived in late 2014 as an evolution of a similar program, AZdecrypt underwent significant enhancements. By early 2016, it was officially released with a Windows graphical user interface (GUI), becoming instrumental in research on the Z340 cipher associated with the Zodiac Killer (Kopal, 2019; Oranchak, 2020). During this project, the software's capabilities expanded with additional solvers and features. Over the years, AZdecrypt has played a key role in decrypting several previously unsolved ciphers (Schmeh, 2021; Vierra, 2023). The software, along with its source code and n-gram data, is freely accessible online.

In order to solve the 1000-letter bigram challenge, Jarl optimized the code of AZdecrypt for deciphering bigram substitutions. The main technique to be applied was hill climbing. The

program's scoring function relied on the frequency analysis of 8-grams (eight-letter blocks). To our knowledge, this marked the first ever application of 8-gram statistics in cryptanalysis.

Since no 8-gram statistics were available, Louie took the initiative to create one. Working with 8-grams presents significant challenges due to their sheer quantity; in a 26-letter alphabet, there are 26^8 possible combinations – approximately 200 billion. Compiling meaningful reference statistics requires an immense volume of text. Louie gathered approximately 2 terabytes of English text sourced from the following materials:

- 1.3 million English public domain books from Project Gutenberg
- 4.5 million potentially copyrighted English books
- All English Reddit comments and submissions
- All of the English Wikipedia from a dump as of 2019-02-25.
- All English Project Gutenberg books
- All of the English subtitles of every movie ever released
- All lyrics of all available English songs ever produced
- 7 billion words from Usenet posts
- 34 million sentences from online news stories
- 135 million online reviews from Amazon, Tripadvisor, and Yelp
- 4.4 million Yahoo Answers exchanges
- Louie's own re-creation of the OpenAI GPT-2 data model

As handling such enormous quantities of data requires a large amount of memory, Louie developed several quirks in how 8-grams are compiled, encoded, and loaded into memory. First, he cleaned up the data with a chi-square filter in order to remove text from other languages. Using Index of Coicidence (IoC) and entropy filters, he removed overly redundant and sparse information. All these methods to clean up the input data are simple and were easy to implement.

Next, the 8-grams were restricted to encodings where both the initial and final four-letter subgrams were valid English language combinations (this included about a quarter of all possible combinations). Each one was given 8-bit capped-log-frequency scores and stored in a compressed pointer table and compressed with the Gzip algorithm.

In addition, Louie developed a novel n-gram format designed to efficiently handle large datasets. The process begins by calculating 4-gram frequencies from the corpus. From these, the top 100,000 4-grams (as an example, based on a chosen metric) are selected. A new table is then created, structured as a two-dimensional array (100,000 × 100,000) with 8-bit values (log probabilities), requiring approximately 10 GB of storage.

This table is indexed using a 4-gram lookup table, which maps each 4-gram to the corresponding values in the main table. The main table is structured as an array $(4g(x_1, x_2, x_3, x_4), 4g(x_5, x_6, x_7, x_8))$. If a given 4-gram has a frequency of zero, its associated 8-gram value is automatically set to zero. This acts as a low-level filter, ensuring that higher n-gram sizes, which are inherently more sparse, are managed effectively.

Another benefit of this system was leveraged by Jarl in AZdecrypt. If the target computer does not have enough RAM available, the table size (100,000 for example) can be adjusted on load time. This allows the n-gram format to work on less powerful machines, albeit with diminished capacities.

Louie's n-gram system is also valuable for lower n-gram sizes. For example, 6-grams can now be made to be much more cache-friendly and therefore run much faster on mainstream computers where the letter n-gram table does not entirely fit in the L3 cache (the L3 cache is the largest cache in current CPU architectures).

The resulting body of text provided 2,062,507,743,806 samples of 8,178,871,377 unique 8-grams. The subgram structure reduced the final file to 3,631,818,052 "valid" English 8-grams, which could be comfortably loaded into 14 GB of memory (instead of 195 GB for a naïve implementation). Jarl and Louie also tried larger models, close to 64 GB, but they observed reduced convergence and slower performance. This happened because many of the $26^4 = 456976$ existing 4-grams (such as XZQQ) don't appear at all, while others (such as XAIB or LMAO) make their way into the raw text data via foreign languages or internet slang. This distorts English 4-gram frequencies if one begins including much beyond 100,000 4-grams. As a consequence, 14 GB turned out to be the best memory size.

To apply hill climbing, AZdecrypt first created two new symbols for every unique bigram based on the following coding method: JOININGTHEJOINTS = 1 2 3 4 3 4 5 6 7 8 1 2 3 4 6 9. Then it changed the key (i.e., the substitution table) one symbol at a time. Bigram homophones (i.e., several ciphertext bigrams decrypting to the same plaintext bigram) were allowed but punished (i.e., they led to a lower result of the scoring function). After a few days of optimization, the program was able to consistently solve the 1346-letter challenge (the one already solved) in less than five minutes even without requiring a crib.

After this success, Jarl felt confident and applied AZdecrypt on the 1000-letter challenge. He started it running before going to work. When he came back, the software had returned a 70-80% accurate decryption in about 4 hours. He shared this result with Louie. They cleaned it up and submitted it to Klaus. Even though it wasn't a 100% solution, it was close enough that it was aceepted.

5 The 750-letter challenge

The next bigram challenge Klaus created was based on a plaintext consisting of 750 letters. The ciphertext is provided in the following (Schmeh, 2019e):

YYXFTVUJKXMYWODAWFZPSAPPVDW
NEXAJXFPPRXKCMFBZIXDLTCVIBSKLZO
XIUKPEMUXFEMDUOGPCRRMWZSVBNM
YYSHLWCIAJJWORCFCHKYRXYYJVUPAG
JHBZAJZPCJSEWZSEWZCJLFWOFHSAEMX
ZZUJHLNGNNMYYIXUVNMYYIXBWAOK
YJRYCHUBMNOQTXAPCRRMWPPWZAML
LPCXFEMWFITKYPGISZEKJMOMUXAERE
KWGQTEOXILBUGGNTCYOYAHUUQZNK
YBJADXIAFICRWCRFPPGZIEEBZHUIWKR
KERRLZWFGQNAJRJQNTPYKBPEKBDLNG
DYXPVAZSSKUVHUBBDLXAWFZUPNHZC
CRXGOLFZUHUGNVWDYRRSAJHTRZUXA
XPKMYYYCHRXZDUQSLFDYKJIAZIDLGQ
NAQXBVWRSWGXPPAJMPDUPPVWAVNA
ORHUUWNBLNFMBSSAPPDVGCGCWFWY
DYZEWOPETHDLMUZURXKJHMKJYUBV
OJWYDYUGCYZPZIDLXFLWPCFSEXZRWF
ERWFIXDTYYWUVJPNAJZURXTFHZOAX
ALZXITHDLBSKLZOXIUKJPYYSHLWCIAJ
XFZURLVWUNPCHUPTXZHCAJANBWLPK
MHUVCWRKXKMBVCXCTHUHMNCQXVB
TCNGADRHPCKWUGRRKBRQXFPGWAMU
DYIXDLKJJSDUOGQTRRKBXILBUGBBIPD
LXZZUWAOSDLYYZPYAZSVBKBGCJPUJX
LLHDYIAKBVBZENMVCRWFA

Again, Jarl and Louie solved the challenge (Schmeh, 2019f). They needed about a week. Here is the plaintext they found:

IN THE FOLLOWING YEAR FOSTER
ACQUIRED THE RIGHT TO USE THE NAME
MARLBOROUGH AND THE MODEL
DESIGNATION NINE HUNDRED WHICH
HAD ORIGINALLY BEEN USED FOR AN
OPEN OPERATING SYSTEM PRESENTED
IN NINETEEN NINETY BY NORWEGIAN
SOFTWARE DESIGNER PETER IDE THE
MARLBOROUGH NINE HUNDRED WHICH

WAS PRODUCED IN A GERMAN FACTORY IN PROBABLY OVER FOUR THOUSAND STYLES IS FAR LESS WELL KNOWN THAN THE LION GAMMA THREE AND THERE ARE SOME AMBIGUITIES AND INCONSISTENCIES REGARDING ITS PRODUCTION NEVERTHELESS IT IS CONSIDERED A MODERN CLASSIC AND STATE OF MASTERPIECE CARS CONNINGHAM THEN DESIGNED A NEW DISCUS CONCEPT FOR THE THUNDERBIRD HARDWARE TRYING TO SOME DESIGN FEATURES FROM THE MARLBOROUGH NINE HUNDRED THESE INCLUDE AN IMPROVED KEYBOARD A NEW COLOR DISPLAY AND ADDITIONAL INPUT DEVICES IN TWO THOUSAND ONE THE NEW MODEL WAS INTRODUCED TO THE PRESS AT THE INFORMATION TECHNOLOGY CONVENTION IN NEW YORK

The plaintext originates from a non-English Wikipedia article, translated into English using DeepL. During the process, Klaus altered all names, locations, technical terms, and numerical data. As a result, the plaintext became entirely unique and had never been published before, rendering it untraceable through online searches.

6 Solution

The program first produced a number of pretty convincing "phantom solves", including one that began: "AT THE FESTIVAL OF THE BULLS…", which scored nearly as well as the eventual solution. Louie spent a day fruitlessly trying to solve the challenge using various strategies along with this false crib.

After several days of no progress, Louie switched to a development version of his 8-gram file that now additionally included all the Twitter (now known as X) data and also some other new extremely large data sources. This model was generally a few percent better in solve accuracy given enough computing time.

After six hours, the program found a solution that was 55.93% letter-accurate, which Louie only noticed after it had been running for about 14 hours. At this point it had become fairly clear that the message the program found was fairly single-topic, so he emailed it to Jarl, believing it was possible that at least some portion of it was correct. Then two hours later, the program refined the result to a 69.49% letter-accurate solution.

Louie helped Jarl crib several portions that were completely unconstrained (with no repeated symbols anywhere in the ciphertext). After a few messages back and forth, the two were able to clean up the solution to a satisfactory result they felt confident in. Nevertheless, there were still a few portions they could not be completely sure of. The "Peter Ide" portion could easily be "Peak Code" or something similar. There was not enough symbol reuse to know for sure.

Jarl and Louie contacted Klaus with their work thus far, and Klaus confirmed the correctness of the solution, which meant that the world record in solving short general bigram substitution ciphertexts had been improved to a ciphertext length of 750 letters.

The main differences between their approach to the 750 versus the 1000 bigram challenge was a slightly more powerful 8-gram model, using more aggressive annealing parameters, some generalized convergence improvements that Jarl added to his program, and a new version of Peter Norvig's word-gram model for automated word division that Louie recently rewrote.

7 The 600-letter challenge

In March 2020, Klaus presented a new general bigram challenge, which as of this writing remains unsolved. It consists of 600 letters (Schmeh, 2020):

UGBZAEHINYQLBPZLNFTLUEBMULTLSL
ZPBZPZKPOVUGYSQPNYHLRYFHATQKR
HTZEHPDQUUGYSUJOVYTUGYVRHAJNF
TLUEXFRUEOOJTZOSLUPZEICVADYMYL
CRBZXOUGSVDJOIDYRHTZOSWZROYNKJ
RMEIXOREOVNFTLUESAMNDJHIIWJGKR
YFUBTIQPULBPRMJORECJCYWZZPQRXX
VNOSZLBLNYJMPLYNOVLCLKIOGUKUKF
SAKAQRSVQXUJIOANYSWZSDKUKFLNR
MEIRJYVEOLXLKMEYKERHXZPBZXOZX

QPCRKSYOSVHNTLIXKRYFUBTIMGWIZL
OSONRMIDKYNYLCFFOMTTLLJHWTADH
LYNRHMZADOGMUKBWZZPPQBZBZNOC
RHINYNFTLUEYNOVBZNOQPGCQMRHTZI
DKYNYCRBZXOUGSVTTQPOSDYXOMQK
KVNEALUYVRMUFPYNXZAVLRHTZNYQ
XMFYVUCMZSAJMBZZPXPBZMNVFUCJT
NYQXGHEITPPYFWKUZFPZQUDEVLDBO
MGRUEKFSCYTVNANLDRMNBYVUTFNUJ
MUMMEOIXIISDVNZPMNRYRCTFUGZPD
NUTLXJNSSVNCRJC

The plaintext is in English. Can a reader break this challenge? If so, they will set a new world record.

8 Conclusion and outlook

The main conclusion of our paper is that it is possible to solve a 750-character general substitution cryptogram using hill climbing. To prove this required significant effort from two of the authors. Their work suggests that solving shorter cryptograms of this kind is likely going to be even more challenging.

It is worth noting that previous challenges of this nature were solved with hill climbing with scoring functions based on 4-gram or 5-gram frequencies. In contrast, the cryptanalysis of the 1000-character and the 750-character ciphertexts relied on 8-gram frequencies. This indicates that hill climbing becomes increasingly effective as the value of *n* in n-gram statistics rises. However, while 9-grams (nine-character blocks) instead of 8-grams might offer even greater potential, finding sufficient amounts of text to produce meaningful statistics remains a significant challenge.

The authors of this work have no proof that the conjecture "the greater *n*, the more powerful are n-gram frequency statistics" is correct for hill climbing and in case it is, why. If a reader knows more about a visible proof, we would be interested to know.

Jarl states "There are practical reasons why greater n-gram sizes may not continue to scale or work well. For example with 10-grams or 12-grams, a unique unseen plaintext may have many 12-grams that are entirely unique to the plaintext. The hill-climber will then cause havoc within this zero space and the 12-grams will lose their strength since they cannot properly discriminate."

The difficulty of breaking general bigram substitutions has to be contrasted with the Playfair cipher, which is a special case of the general bigram substitution (Dunin and Schmeh, 2023). As mentioned, a Playfair ciphertext with only 26 letters of ciphertext can be solved. Comparing that with the difficulty of solving the general bigram substitution, we can conclude that the latter is by two orders of magnitude more secure than the Playfair. It is noteworthy that an even shorter Playfair ciphertext is solvable with other techniques such as a dictionary attack, but this is not applicable on the general bigram substitution, provided that the substitution table is generated at random. On the other hand, a general bigram substitution requires a large table and is therefore considerably more difficult to handle than a Playfair.

While the 750-letter challenge was solved, as of this writing in 2025, the 600-letter variant has remained unbroken for four years. This suggests that the limit may have been reached. A detailed evaluation of this question based on confidence limits (Kubáček 1994) is not within the scope of this work.

Jarl and Louie state that they are working on the 600-letter challenge. Louie has proposed a more efficient hill-climbing approach by replacing the current approach that includes homophones with a true one-to-one bigram substitution solver. Following this suggestion, Jarl developed a proper one-to-one bigram substitution solver, which has shown significantly better performance compared to the original method. Additionally, Louie has compiled new 8-gram statistics using a considerably larger dataset. The amount of text to produce these statistics has grown from 2 terabytes to 10 terabytes. The additional 8 terabytes of data came primarily from more Reddit data (produced from 2020-2024) and ParaCrawl (English ParaCrawl data is also a processed form of CommonCrawl) (Figure 2).

In addition, Louie improved the pre-filtering, cross-weighting between different corpora, and the use of dynamic range in model representation. With these improvements, data like Bitcoin addresses or long strings of non-English text can still be used, which will help lead to more accurate frequency counts.

Meanwhile, Jarl for his part has been focusing on compressing n-grams using neural networks with binary weights optimized for fast CPU retrieval. He hopes this approach will complement Louie's n-gram system, particularly for larger n-grams, as Louie's system already effectively handles unseen n-grams.

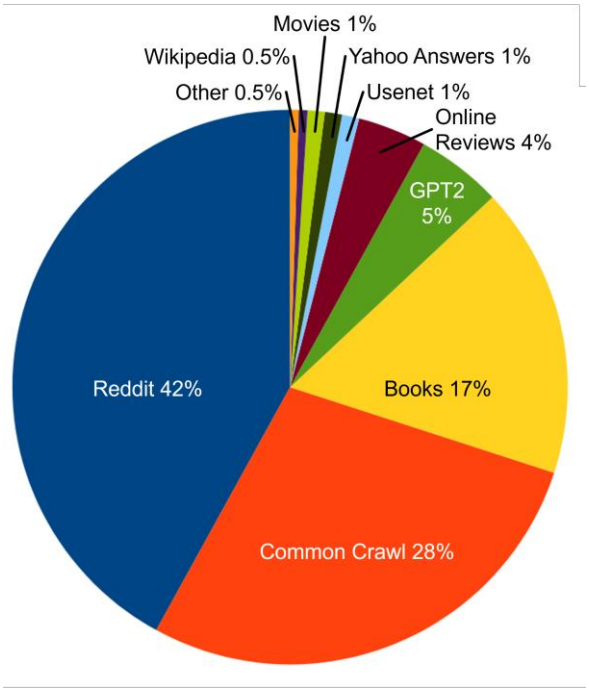

Figure 2. The amount of text used by Louie to generate 8-gram statistics has grown to 10 terabytes. The sources are shown in this diagram.

The enhancements currently being made by Jarl and Louie to their software tools may ultimately pave the way for solving the 600-letter challenge.

Acknowledgments

The authors would like to thank Richard Bean, Norbert Biermann, Thomas Bosbach, Thomas Ernst, Nils Kopal, Armin Krauß, George Lasry, Dave Oranchak, and Christoph Tenzer.

References

Elonka Dunin, Magnus Ekhall, Konstantin Hamidullin, Nils Kopal, George Lasry, Klaus Schmeh. 2021. *How we set new world records in breaking Playfair ciphertexts*. Cryptologia Volume 46, 2021 (4): 302-322

Jarl Van Eycke. 2016. *AZdecrypt 1.22*. https://forum.zodiackillerciphers.com/community/zodiac-cipher-mailings-discussion/azdecrypt-1-19b/

Elonka Dunin, Klaus Schmeh. 2022. *Codebreaking: A Practical Guide*. No Starch Press, San Francisco: 245-247

David Kahn. 1996. *The Codebreakers*. Scribner, New York: 138

Nils Kopal. 2019. *Cryptanalysis of Homophonic Substitution Ciphers Using Simulated Annealing with Fixed Temperature*. https://ep.liu.se/ecp/158/012/ecp19158012.pdf

Lubomir Kubáček. 1994. *Confidence limits for proportions of linguistic entities*. Journal of Quantitative Linguistic Volume 1. 1994: 56-61

David Oranchak. 2020. *The 340 Is Solved!* https://www.youtube.com/watch?v=-1oQLPRE21o

Klaus Schmeh. 2017. *Bigram substitution: An old and simple encryption algorithm that is hard to break*. https://scienceblogs.de/klausis-krypto-kolumne/2017/02/13/bigram-substitution-an-old-and-simple-encryption-algorithm-that-is-hard-to-break/

Klaus Schmeh. 2019a. *Can you solve this bigram challenge and set a new world record?* https://scienceblogs.de/klausis-krypto-kolumne/2019/07/13/can-you-solve-this-bigram-challenge-and-set-a-new-world-record/

Klaus Schmeh. 2019b. *Norbert Biermann solves bigram challenge and sets a new world record*. https://scienceblogs.de/klausis-krypto-kolumne/2019/08/13/norbert-biermann-solves-bigram-challenge-and-sets-a-new-world-record/

Klaus Schmeh. 2019c. *Solve this bigram challenge and set a new world record*. https://scienceblogs.de/klausis-krypto-kolumne/2019/10/07/solve-this-bigram-challenge-and-set-a-new-world-record/

Klaus Schmeh. 2019d. *Bigram 1000 challenge solved, new world record set*. https://scienceblogs.de/klausis-krypto-kolumne/2019/10/27/bigram-1000-challenge-solved-new-world-record-set/

Klaus Schmeh. 2019e. *Solve this bigram challenge and set a new world record*. https://scienceblogs.de/klausis-krypto-kolumne/2019/12/12/solve-this-bigram-challenge-and-set-a-new-world-record-2/

Klaus Schmeh. 2019f. *Bigram 750 challenge solved, new world record set*. https://scienceblogs.de/

klausis-krypto-kolumne/2019/12/19/bigram-750-challenge-solved-new-world-record-set/

Klaus Schmeh. 2020. *Solve the Bigram 600 challenge and set a new world record*. https://scienceblogs.de/klausis-krypto-kolumne/2020/03/03/solve-the-bigram-600-challenge-and-set-a-new-world-record/

Klaus Schmeh. 2021. *Jarl Van Eycke löst 400 Jahre alte Längengrad-Botschaft*. https://scienceblogs.de/klausis-krypto-kolumne/2021/02/18/jarl-van-eycke-loest-400-jahre-alte-laengengrad-botschaft/

David Vierra. 2023. *Solutions to Feynman Ciphers #2 and #3*. https://codewarrior0.github.io/cipher-blog/2023/05/27/feynman-solved.html