

Decipherment as Regression

Solving Historical Substitution Ciphers by Learning Symbol Recurrence Relations

Nishant Kambhatla, Logan Born, Anoop Sarkar

May 2023

Findings of the Association for Computational Linguistics: EACL 2023

Presented by Morten Munk Andersen

Contents

- 1. Why this paper? 2
- 2. Methodology 4
 - 2.1 Recurrent Integer Sequences 5
 - 2.2 Generative Decipherment Model 7

1. Why this paper?

1. Why this paper?

Relevancy

- Homophonic substitution ciphers

Ranking

- Core2023 Ranking: A

Recency

- May 2023

Decipherment as Regression: Solving Historical Substitution Ciphers by Learning Symbol Recurrence Relations

Nishant Kambhatla Logan Born Anoop Sarkar
School of Computing Science, Simon Fraser University
8888 University Drive, Burnaby BC, Canada
{nkambhat, loborn, anoop}@sfu.ca

Abstract

Solving substitution ciphers involves mapping sequences of cipher symbols to fluent text in a target language. This has conventionally been formulated as a search problem, to find the decipherment key using a character-level language model to constrain the search space. This work instead frames decipherment as a sequence prediction task, using a Transformer-based causal language model to learn recurrences between characters in a ciphertext. We introduce a novel technique for transcribing arbitrary substitution ciphers into a common *recurrence encoding*. By leveraging this technique, we (i) create a large synthetic dataset of homophonic ciphers using random keys, and (ii) train a decipherment model that predicts the plaintext sequence given a recurrence-encoded ciphertext. Our method achieves strong results on synthetic 1:1 and homophonic ciphers, and cracks several real historic homophonic ciphers. Our analysis shows that the model learns recurrence relations between cipher symbols and recovers decipherment keys in its self-attention.¹

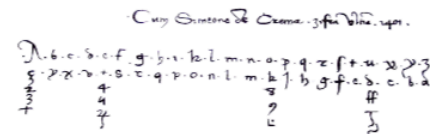


Figure 1: The homophonic substitution key for the *Simeone de Crema* written in Mantua in 1401 AD. The top line maps each character in the alphabet to its reversed-alphabet equivalent; each vowel is substituted by three additional symbols.

sequences (D’Ascoli et al., 2022). We rethink decipherment as a regression task that predicts a natural language plaintext by learning a recurrence relation between integer-coded ciphertext symbols.

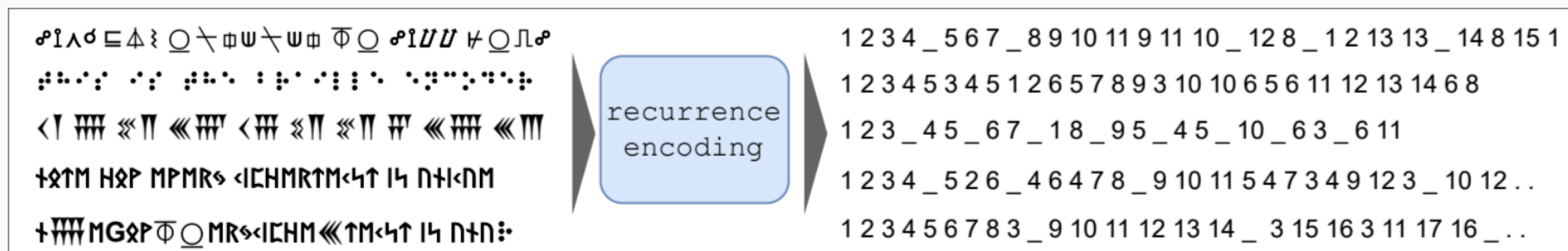
There exist large collections of historical ciphers (see de-crypt.org)², in the form of encrypted letters and more informal communications, of which many remain undeciphered. Many of these texts employ complex *homophonic substitution ciphers*, which mask the frequencies of letters by using a larger alphabet than the underlying language. Figure 1 shows the first known homophonic cipher from 1401 AD³. Automated computational deci-

2. Methodology

2.1 Recurrent Integer Sequences

Capturing first/repeated symbol occurrences

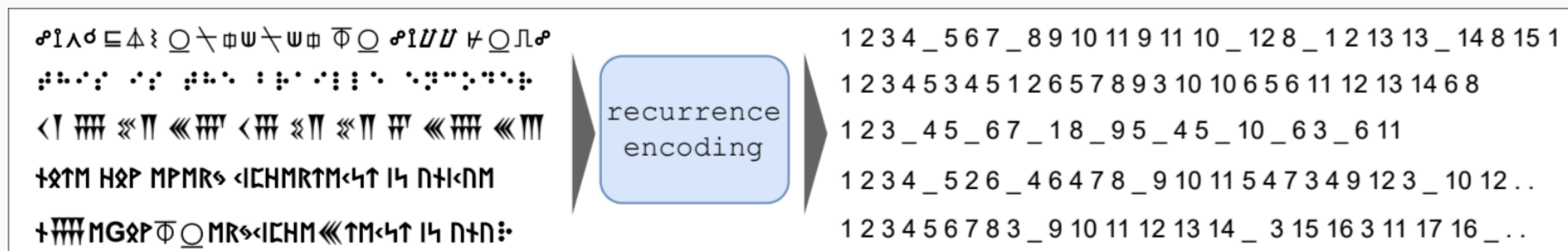
- Spaces denoted as **underscore**
- Unseen symbols denoted as **incremental integer**
- Recurring symbols denoted as represented **previous integer**
- Works for ciphers with different symbol sets



2.1 Recurrent Integer Sequences

Capturing first/repeated symbol occurrences

- Spaces denoted as **underscore**
- Unseen symbols denoted as **incremental integer**
- Recurring symbols denoted as represented **previous integer**
- Works for ciphers with different symbol sets



The authors consider this a novel approach

2.2 Generative Decipherment Model

Remember: Ciphertext is now a Recurrent Integer Sequence

CausalLM

- From left to right - can only look back
- Past words affect predicted words - (sort of like autocorrect)

Dataset made by authors

- 2 million unique homophonic substitution ciphers
- Including corresponding plaintexts
- Modern English

2.2 Generative Decipherment Model

test