Can Artificial Intelligence Solve the Mysterious Anagram From the Church of the Poor Clares in Bratislava?

Eugen Antal
Slovak University of
Technology in Bratislava,
Slovakia
eugen.antal@stuba.sk

Pavol Zajac
Slovak University of
Technology in Bratislava,
Slovakia
pavol.zajac@stuba.sk

Abstract

A mysterious anagram was found in the Church of the Poor Clares in Bratislava, but as far as we know it has never been successfully solved. The anagram contains 81 symbols, including specific diphthongs AE, CH, and GY. Unlike other anagrams typical of that age, the symbols are not ordered alphabetically. We suspect that a specific order of symbols is related to the original order of symbols in the plain text. Even with the suspected order of letters, the number of possible plain text candidates is too high to obtain the original text with standard methods. We examine alternative scoring methods based on modern AI text similarity to improve the quality of the candidate plain text candidates.

1 Introduction

This article focuses on a mysterious anagram related to the Order of Poor Clares. In the following sections, we briefly introduce the main historical facts about Poor Clares and the Church of the Poor Clares in Bratislava, which may be related to the anagram. We present the details and analysis of the anagram and approach how we tried to solve the anagram. We also discuss various artificial intelligence (AI) methods we believe can lead to solving this mystery.

2 Poor Clares in Bratislava and a Mysterious Cryptogram

The Poor Clares (the second Franciscan branch, also called the Order of Saint Clare) is an enclosed order of nuns in the Roman Catholic Church. They came to Bratislava from Italy in the thirteenth century. In 1297 Andrew III of Hungary donated (Szyllaba, 1944) an abandoned church and monastery to Poor Clares. They started to use them later at the end of the thirteens century after a reconstruction. In the second half of the fourteenth century, a church tower was built. The monastery burnt down in 1515, the church was also damaged by fire in 1590. The reconstructions started in 1637 by Péter Pázmány and were finished in 1640 by Losy Imre (Szyllaba, 1944).

In 1529 the Poor Clairs had to escape from Bratislava because of the Ottoman Empire's (close) siege of Vienna. None of the previous nuns returned to Bratislava. In 1541 new nuns of the order of Poor Clares were sent from Óbuda to Bratislava (Szyllaba, 1944). The nuns brought with them various treasures (gems, gold, etc.) and valuable relics (statue, altar, head of Queen Elizabeth, and other remains, Elizabeth's golden crown, etc.) of Queen Elizabeth - the mother of Louis I of Hungary (Louis the Great). In 1618 the nuns of the Dominican Order came to Bratislava and were merged with the Poor Clares (Szyllaba, 1944). These nuns also didn't come with empty hands. They brought treasures (altar, golden chalice, Saint Mary's picture with gems), and valuable relics (gold plated silver box with the head of Saint Margaret, and other remains) of Saint Margaret of Hungary. In 1660 a Saint Margaret statue was donated by György Lippay to Poor Clares. In 1714 some of the nuns moved to Buda. They took the part of treasures and relics with them (Szyllaba, 1944).

In 1618 a church bell was created by Georgius (György) Arnold and installed in the church. In 1700 the church tower was damaged by an earthquake. The Church tower was reconstructed by Abbess Balassa Eva Borbala in 1702.

The Order of Poor Clairs was dissolved in 1782 by Joseph II, Holy Roman Emperor. Most of the remaining relics and treasures were taken away. Based on Szyllaba (1944), in 1782 József Dankó a bricklayer from Bratislava was entrusted to empty

a place in one of the crypts[1], where Balassa Eva Borbala was buried. Several relics, documents, and books were hidden there. Later the bricklayer also carefully walled it. Unfortunately, there are no available direct archival documents to prove this information.

After 1782 the church was used by other people, also for other purposes such as a library (Szyllaba, 1944) or as a museum (Fiala, 2001). In 1851 the church was renovated. In 1895 various fundraising events (also a sound competition by Frigyes Dohányi and Mózes Gaál) were held for the restoration of the pipe organ. Archduke Friedrich (Frigyes) and his wife Isabella also donated money to the Church of the Poor Clares in Bratislava (Szyllaba, 1944).

In 1901 a copper plate was found (Szyllaba, 1944) in the crypt of the demolished sacristy. This plate contains a cryptogram, the content of which is still unknown. Based on (Fiala, 2001) the plate could be dated to the eighteenth century. However, there was no official date estimation performed. It can be connected to donations or reconstructions of the church and also to the treasures hidden in the crypt.

2.1 Copper Plate With the Cryptogram

The copper plate that was found in the crypt (see Figure 1 and the Appendix) consists of 84 letters, from which the first three letters D.O.M. - the initial heading most probably stands for the phrase *Deo Optimo Maximo*. The cryptogram therefore consists of the remaining 81 letters, split into 9 rows and 9 columns.

Some letters are diphthongs and are specially printed to be able to distinguish them from other letters. The letters are ordered specially (see Section 3 for more details). Based on the letter ordering and on the letter frequencies (Section 4) we assume that the mysterious message is a Latin anagram. Please note, that some diphthongs such as GY are not common Latin, however, they can be also part of Hungarian names (such as György, Frigyes directly related to the church, and Poor Clares) or Hungarian city names. If we assume that the rumors about hiding the treasures in the crypt where the Abbess Balassa Eva Borbala was buried are true, we can guess some parts of the plain text to contain names and titles, e.g.: Balassa Eva abbatissa (which could explain some deviations in letter frequencies). It is not clear, whether the plain text also contains a date in Roman numerals. If there is a date, it cannot contain the letter X. A potential candidate is the Church tower reconstruction year, 1702 encoded as MDCCII can be constructed from the letters of the anagram.

The copper plate is currently deposited in the Bratislava City Museum, with number F1-2819. The material of the plate is copper. It is a thin plate with approx.[2] height 164 mm and width 108 mm. The weight of the plate is 120 grams. More detailed statistics of the anagram are presented in Section 4.

Figure 1: The mysterious anagram (From the collections of Bratislava City Museum, Slovak Republic)

3 Preliminaries and Notation

Let $\mathbf{m}$ denote a sequence of N letters $(m_1, m_2, \ldots, m_N)$ from alphabet A. We will call it an original phrase in short. In the examples, we will typically write the letter with punctuation given by spaces, but from the mathematical and computational point of view, spaces are ignored.

Let us denote the index set $I = \{1, 2, \ldots, N\}$. Let π be a permutation on I. Anagram of $\mathbf{m}$ is

[1] The church and the monastery had a total of three crypts.

[2] The plate does not have the same width (also height), there is a difference of a few millimeters.

any sequence $\mathbf{m}_\pi = (m_{\pi(1)}, m_{\pi(2)}, \ldots, m_{\pi(N)})$. Let $\leq$ denote a natural order of letters in the alphabet A. An anagram $\mathbf{m}_\pi$ is normalized, if $m_{\pi(i)} \leq m_{\pi(j)}$ for each $i \leq j$. Note that if some letters in $\mathbf{m}$ are repeated, there are multiple permutations π that produce the normalized anagram of $\mathbf{m}$.

To solve the anagram, we want to recover the original phrase, given $\mathbf{m}_\pi$. However, due to the composition of permutations, any $\mathbf{m}_{\pi'}$ can be potentially considered a valid solution of the anagram unless we have a method that can identify the original $\mathbf{m}$. The identification can be based e.g. on the grammar structure of the message, or related to a specific meaning of the message. We will use the term "anagram solving" in a weaker sense: our goal is to find any $\mathbf{m}_{\pi'}$, which satisfies some criteria, e.g. syntactic and semantic consistency with a target language and/or expectations of the meaning of the message.

For each sequence $\mathbf{m}$, we can define a function $pos_{\mathbf{m}} : A \to I \cup \{\infty\}$ as follows: If $x \in \mathbf{m}$, $pos(x) = \min\{i;\ m_i = x\}$, otherwise $pos(x) = \infty$. Any sequence $\mathbf{m}$ thus defines a specific order $\leq_{\mathbf{m}}$ of letters from the alphabet A: $x \leq_{\mathbf{m}} y$ is and only if $pos_{\mathbf{m}}(x) \leq pos_{\mathbf{m}}(y)$. An anagram $\mathbf{m}_\pi$ is ordered, if $m_{\pi(i)} \leq_{\mathbf{m}} m_{\pi(j)}$ for each $i \leq j$.

Simply said, the ordered anagram preserves the order of letters in the original phrase. An example of the ordered anagram of the phrase `MYSTERIOUS ANAGRAM` is `MMYSSTERRIOUAAANG`. In the ordered anagram, the prefix of the original phrase is partially visible, which can help in solving the anagram. Moreover, if we consider ordered anagrams, the space of potential solutions is reduced, because only some of the $\mathbf{m}_{\pi'}$ are ordered.

4 The Analysis of the Anagram

The first step of the solving process involves the study of the material and preliminary analysis. The anagram consists of $N = 81$ symbols, out of which $m = 20$ are distinct. Namely, the symbol counts n_i (in order) are:

```
{'p':  2, 'l':  3, 'v':  6, 'm':  4,
 'b':  5, 'e':  7, 'a': 12, 'c':  3,
 's':  7, 't':  8, 'n':  4, 'o':  3,
 'i':  5, 'ch':  1, 'r':  3, 'ae':  3,
 'd':  2, 'f':  1, 'gy':  1, 'h':  1 }
```

The number of symbols is too low to determine the plaintext language with statistical significance. However, from the context of the anagram, we suspect that the language of the inscription is (medieval church) Latin. This hypothesis is based on the artifact location and dating, as well as the initial heading D.O.M. (Deo Optimo Maximo). Inscription features special diphthongs AE, CH, and GY, with only AE being native to Latin. We suspect that CH and GY can either be local spellings of original Latin words (e.g. CHRISTUS, AEGYPTUS). Another hypothesis is that these letters are parts of the names of places or people.

4.1 Key Space

If the order of symbols is not significant, it is possible to construct C anagrams from these symbols, where:

$$C = \frac{N!}{\prod_{i=1}^{m}(n_i!)} = \frac{81!}{2! \cdot 3! \cdots 1!} \approx 2^{285}.$$

The formula for C is known: we count all possible orders of all symbols (numerator) and divide out the orders of repeated letters (denominator).

If the order of symbols is significant, the formula can be derived recursively as

$$C_{j-1} = C_j \cdot 1 \cdot \frac{\prod_{k=1}^{n_j-1}(k + \sum_{i=j}^{m} n_i)}{(n_{j-1}-1)!}.$$

Here C_j is a number of possible sequences of symbols from j-th symbol up to m-th. We must to prefix these sequences with a $(j-1)$-th symbol (only 1 choice). Then we can place any of the remaining $n_j - 1$ of the same symbols between already placed $(k + \sum_{i=j}^{m} n_i)$ symbols. Because of repetitions, we need to reduce this count by a factor of $(n_{j-1}-1)!$ (the number of repeated symbols except the first one on a fixed position).

In comparison to the general anagram, we are restricted in the placement of the first symbol in each group. The closed (non-recurrent) formula thus becomes

$$C_1 = \frac{N!}{\prod_{j=1}^{m}(n_i!)} \cdot \frac{\prod_{j=1}^{m-1} n_j}{\prod_{j=1}^{m-1}(\sum_{i=j}^{m} n_i)}$$

Thus, under the hypothesis that the order of symbols in the original anagram is significant, the search space is $C_1 \approx 2^{232}$. This space is thus too large to be efficiently enumerated and computationally explored by exhaustive search.

5 Basic Anagram Solver

In order to solve the anagram, we need more sophisticated methods than an exhaustive search.

One option is to use meta-heuristic algorithms that explore the space of 2^{232} symbol permutations and try to maximize a likeness to Latin text using some suitable fitness function.

An alternative approach is to focus on complete words and phrases instead of symbols. We try to concatenate textual elements that can be contained in the original anagram while verifying the validity of the solution. Basic validation is provided by the order and number of used symbols. An intelligent search should also focus on textual criteria such as Latin grammar and textual meaning.

5.1 Text Preparation

The preparatory phase for the search is the identification of individual words and short phrases that can be a part of the anagram.

In the initial experiments, we use a simple corpus obtained from two sources:

1. Latin Bible translation (Vulgata) (Bible, 1976) — this was used for initial experiments due to the potential religious context of the anagram,

2. Georgius Agricola : De re metallica/Liber VII (Agricola, 1556) — this is due to the fact that the text potential starts with the word PLVMBVM (plumbum, lead), and a potential partial decryption used in Google search pointed us to Agricola's treatise,

3. The Bull of Pope Alexander VI on the cannonization of St. Clare of Assissi (Pope Alexander IV, 1255) — for Latin terms related to St. Clare,

4. List of personal Hungarian names from 16th century (van Nijmegen, 2002), and surnames of Hungarian nobility (de Zepetnek, 2010) — to have candidate names for the suspected author of the anagram. In our experiments, we only use names that contain the 'GY' digraph.

Before the search, the text was processed and split into individual words and short phrases (to prevent meaningless small words from complicating computer search). The processed phrases were then stored in a dictionary-like data structure for easier processing during the search.

5.2 Beam Search

It is computationally impossible to examine all possible (ordered) anagrams of our cipher text. Instead, we can use an intelligent search. The first algorithm we tried is greedy depth-first search with cut-off (a beam search): Start from some phrase, identify potential continuation, score them, and then continue recursively with the continuation candidate that has the highest score.

During the search, we keep track of the current partial phrase, remaining fixed-order symbols, and free symbols. Fixed-order symbols can only be used in a given order, while free symbols can be combined in any order. We have a list of words and partial phrases we evaluate as potential continuations of the given partial phrase. First, we check whether the new partial phrase can be concatenated with the previous one while remaining a sub-anagram of the target. For each potential continuation, we again obtain new remaining fixed order symbols and free symbols and assign a score that is used in the search.

The score assigned to the continuation candidate should reflect its suitability to help the final search. In the basic version of the search, we try to score continuation candidates relative to the purposes of reconstructing the ordered anagram, regardless of their meaning. After preliminary experiments, our score consists of three components:

1. if the candidate uses a new fixed order character, we assign it a flat bonus of 20 points (to try to fix the anagram order as soon as possible),

2. each character of the candidate counts as 1 point (preference to longer continuous continuations),

3. each character that remains in the set of free characters gives 10 points to the score (preference to more freedom in latter search stages).

Typically, we have a high number of potential continuation candidates in each step. To reduce the complexity, we only examine at most a fixed number $N = 10$ of candidates on each level (a so-called beam search), in the order based on their score. If the number of candidates is higher, we select N candidates with a roulette selection, where the probability of selection is directly proportional to the score. If there are no viable continuation

candidates, we use a simple backtrack. Similarly, we backtrack a level, if we examine N candidates at a given level.

5.3 Experimental Results

The beam search described in the previous section can successfully identify ordered anagrams of the target anagram on a single PC within a few minutes. Typically, the algorithm can cover almost the whole phrase up to a small number of letters that cannot be combined further and require excessive backtracking. In such a case, we simply restart the algorithm, but depending on the parameter N (branching of the beam search), we can just let the search finish on its own. The character of the search indicates, that a stochastic search, such as hill-climbing or evolutionary heuristics, would also be suitable for the purpose.

The textual quality of the found ordered anagrams is not good and reflects the selection of the corpus material. Examples of the solutions provided by the simple search:

- `Plumbum eum paululum castas et Saba necnon tibi et Abisai Nachor eboris caedet Arfad testae Agy hastae`
- `Plumbum levam clama est natus putabant assensu se nobis iacio Bebai charta Cretae additae Fogy Aether`
- `Plumbum melle accessu suas et nomina et citato es satiata peius Chobar barbae tradi fundae Bagy Aethan`

We remark that our goal is to estimate the feasibility of solving a specific anagram. A more thorough systematic future research of the proposed methods is needed to evaluate their overall suitability to solve ordered anagrams in general, and to optimize the parameters.

6 Anagram Solver Enhanced With AI Text Scoring

The critical part of the algorithm from Section 5 is the computation of the score for the candidate. The original score tries to maintain the largest possible freedom of choice for the continuation candidates and prefers longer phrases (from the corpus) to accelerate the search. However, grammar structure or meaning does not affect the score.

The rapid progress in natural language processing (NLP) gives us ready-made tools to use in text evaluation and generation. Language models can be used to predict continuations of phrases and evaluate the consistency of existing (partial) phrases. The main problem in our research is to connect NLP processing in an efficient way with a complex stochastic search within a large space of potential anagrams. If we focus on text generation, the pre-trained predictive language model focuses on text predictions that are not related to our anagram. It might be possible to train a custom language model conditioned on our anagram symbol distribution, but we believe that this would require a dedicated research and a large number of computing resources.

For our purposes, we focus on simpler language models that can evaluate text structure and language similarity. NLP models can analyze the structure of the sequence, and identify grammar sections such as noun phrases or verb phrases, as well as grammatical categories (number, case, gender, etc.). Language similarity score can identify meaningful continuations based on the similarity of the extended phrase with its predecessor. We can also use a language similarity score to compare the partial candidate phrase with some hypothetical phrase we suspect is encoded in the target anagram.

There are three main places of the NLP tools in our anagram solving:

- In the pre-processing phase we can analyze the potential corpus. We can identify larger meaningful blocks of text that can be used as candidates for continuation (such as noun phrases, and verb phrases). We can also add extra details to candidate sub-phrases (such as grammatical categories) that can be taken into account when scoring continuations.
- Directly during the continuation candidate scoring. Note that we need to evaluate a large number of continuation candidates even during the restricted beam search, thus we need to use only very basic and fast NLP functions in this phase of the search.
- In the final evaluation of the candidates. If our search algorithm can quickly provide a

large number of potential whole ordered anagrams of the target anagram, the NLP tools can then evaluate the result and identify the most promising plain texts.

In our experiments, we use the language model `la_core_web_lg` (Explosion, 2023), with the `spaCy` library (Burns et al., 2023) that provides NLP functionality.

6.1 NLP Preprocessing

The main problem with using NLP for corpus processing is that medieval Latin is a very complex language. It is not clear how to properly incorporate the grammar into scoring functions, as Latin is quite flexible with word order, and the meaning can be significantly changed by declensions. E.g. a possible start of the sentence in our anagram can be both 'plumbea capsula' (nominative), but also 'plumbeam capsulam' (accusative). The choice of the declension in the first part of the phrase will then influence the rest of the sentence structure.

It is also not clear whether the plain text is a single sentence, multiple sentences, or whether it is even written with a proper grammar structure. Construction of the algorithm that takes Latin grammar into account would require the involvement of a language expert, and it might be easier to just construct a suitable anagram by hand.

Even if we were to take Latin grammar into account, it does not guarantee that we produce a meaningful result. As an example, we have created a simple modification of our basic algorithm that alternates 'noun phrases' and 'verb phrases' (as identified by NLP processing of the corpus). Under these restrictions, the algorithm can still find the ordered anagram of the target, with a similar performance to the basic algorithm (but with a longer pre-processing phase). Unfortunately, the results are still non-sensical:

- `plumbum evacues Asub captum eas tenentem nocte solito lis sana Achis bibi Abrae datae far gyrat hae dat`
- `plumbum evacues Paulus metent colono mactatis Ananias bibe Becher stabat taedae risi fas gyrat hae da`
- `plumbum evacues aucupes celata notos meos mansi Achias intrant Bebet bibit Laed Safat aer gyrat hae da`

6.2 NLP in Candidate Scoring

In our case, a tight beam search (with low N) provides only a few candidates for the whole ordered anagram. However, if we set N higher, the search takes too much time and produces a large number of candidates with a similar beginning. It is thus more suitable to focus on using AI scoring already during the search, and not leave it to the end.

We estimate the language quality of the continuation candidates with the use of just the function for language similarity. The original idea was to use language similarity between the examined candidate and its extension. We start the process by selecting a suitable phrase prefix manually and then follow the text similarity score (the N best candidates, without the roulette selection). Unfortunately, this scoring system often leads to repetitions. Furthermore, the scoring based on a language model does not properly handle proper names (Biblical figures, Hungarian names, ...).

An example solution provided by using a simple beam search following the text similarity score:

- *`plumbea capsula`* `massa massa massa Becbuc Tabeel Benennon Atita Choo Iturei taeduit Frigy taedae Ruth`

An alternative scoring system can be based on a text similarity of the partial anagram to some hint phrase (context). The authors of this article do not know Latin language. However, we can still use LLM AI assistants for machine translation of English prompts, along with phrase and word suggestions and grammar consultations. In our experience, the main advantage of AI assistants is that they can quickly provide and modify hint phrases without excessive search of dictionaries and thesauri. However, the AI assistants lack logic and are not trained in related tasks. E.g., the assistants were repeatedly failing to provide words containing restricted sets of letters, or letters in some prescribed order. However, due to the exploratory nature of our research, we have not conducted systematic experiments in the area.

An example of the hint phrase suggested by artificial intelligence (under our guidance) is the phrase `Capsula plumbea cum sacris reliquiis in sacristia abscondita est. Laus tibi, Christe Iesu.` Using language

similarity to this phrase, our search algorithm provides the following result:

- *plumbea capsula tenet* velet committant et Sabias sanat sociabis Schaur Boma aeneae Adae Dub Frigy Hur

The structure of the anagram is better, but the phrase is again meaningless. The ending of the phrase is saturated with names from our list of Hungarian nobility names.

7 Conclusions

Artificial intelligence is rapidly evolving and can help in tasks related to natural language processing even for researchers that are not familiar with the language. However, in the current state, there are still some tasks that require an expert and cannot be easily automated even with the pre-trained language models.

One of the tasks that AI assistants are good at is the recognition that a given anagram does not form a meaningful phrase. Thus, our mysterious anagram can potentially be solved by AI by feeding final phrases to language analysis. However, there are too many candidates, even if we restrict the search to phrases from the corpus. In our manually conducted search with a larger corpus from the Latin library collection (Carey, 2023) (obtained through a corpus importer of the Classical Language Toolkit (Johnson, 2021)), there are 4 suitable starting words of the anagram ('plumbum', 'plumbeum', 'plumbea', 'plumbeam'), and an abbreviation ('p.p.'). However, there are 5023 possible continuations of 'plumbea', 68415 continuations of 'plumbea capsula', and more than 200 thousand continuations of 'plumbea capsula tenet' (with only 19 out of 81 anagram symbols used). A simple evaluation of language similarity for these continuations takes more than 2 hours on a PC.

A more viable strategy might be the application of an algorithm that works by improving, connecting, and correcting several parts of the text with semantics and context. We have done this process manually, producing a working hypothesis:

- *plumbea capsula tenet*
  vvvvmmeeeaaasttnnn oooi
  CHartAE abbatissAE balassAE
  rr fGYh
  abdita est, MDCCII

Symbols 'o' and 'i' must be used before 'CHartAE' in that order, similarly symbols 'f','GY', and 'h' before the ending. The hypothesis has the structure: *leaden box that contains* (something related to) *documents of abbatissa Balassa* (place of hiding) *were hidden, 1702*. Notice that this process requires a lot of backtracking and changes, and as we cover more letters in the anagram, it becomes harder to correct mistakes or construct meaningful words from the remaining symbols.

Even though the space of possible (ordered) anagrams is very large, with the help of Latin experts it might be possible to recover the original phrase (or at least a correct Latin phrase that makes sense in the correct context). Employment of artificial intelligence for this task might be possible, but it would require constructing and training a custom model that can handle specifics of anagram solving (such as taking into account the number and order of symbols, and specific historical context).

Acknowledgments

This research was supported by grants VEGA 1/0105/23, and VEGA 2/0054/24.

References

Georgius Agricola. 1556. *De re metallica/Liber VII*. Wikimedia Foundation. https://la.wikisource.org/wiki/De_re_metallica/Liber_VII.

Bible. 1976. *Vulgate*. Oxford Text Archive. https://ota.bodleian.ox.ac.uk/repository/xmlui/handle/20.500.12024/0319.

Patrick J. Burns, Nora Bernhardt, Tim Geelhaar, and Vincent Koch. 2023. spaCy Project: la_core_web_xx. https://github.com/diyclassics/la_core_web_lg.

William L. Carey. 2023. The latin library. http://thelatinlibrary.com/about.html.

Steven Tötösy de Zepetnek. 2010. *List of Historical Surnames of the Hungarian Nobility*. Library Series, CLCWeb: Comparative Literature and Culture. http://docs.lib.purdue.edu/clcweblibrary/nobilitashungariae.

Explosion. 2023. Industrial-strength natural language processing in python. https://spacy.io/.

Anton Fiala. 2001. Tabuľka záhadnosti v zbierkach mestského múzea v bratislave. *Múzeum*.

Kyle P. Johnson. 2021. Classical language toolkit. http://cltk.org/.

Pope Alexander IV. 1255. *Clara claris praeclara*. Franciscan Archive. https://franciscan-archive.org/bullarium/clara.html.

Dr. Emil Szyllaba. 1944. *A pozsonyi Klarissza-templom története*. Bratislava : Litera könyvnyomda és könyvkiadóvállalat.

Walraven van Nijmegen. 2002. *Hungarian Personal Names of the 16th Century*. Society for Creative Anachronism, Inc. https://heraldry.sca.org/names/hungarian/.

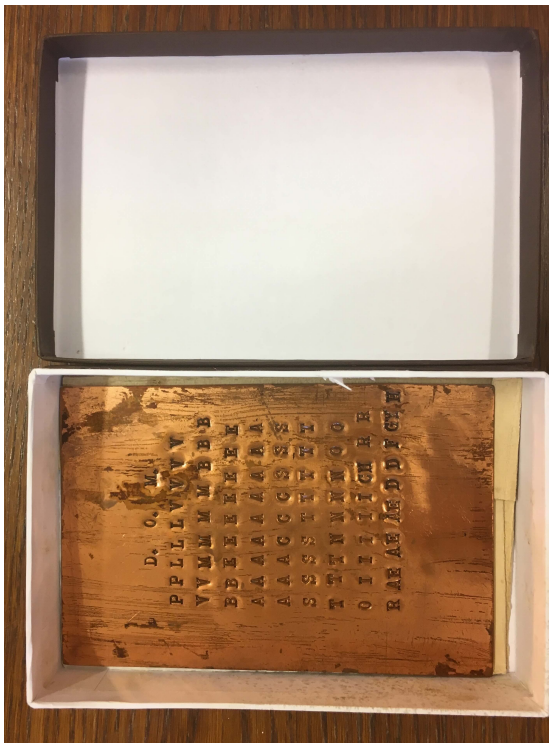
Appendices

Figure 2: Copper plate with the anagram (From the collections of Bratislava City Museum, Slovak Republic)

(a) Front side

(b) Back side

Figure 3: Front side and back side of the plate (From the collections of Bratislava City Museum, Slovak Republic)



(a) Deposited in a box

(b) Weight measurement

Figure 4: More pictures of the plate with the anagram (From the collections of Bratislava City Museum, Slovak Republic)