Supporting Historical Cryptology: The Decrypt Pipeline

Mihály Héder[1], Alicia Fornés[2], Nils Kopal[3], Ferenc Szigeti[1] and Beáta Megyesi[4]
[1]Budapest University of Technology and Economics, Hungary
[2]Autonomous University of Barcelona, Spain
[3]University of Siegen, Germany
[4]Stockholm University, Sweden

Abstract

We present a set of resources and tools to support research and development in the field of historical cryptology. The tools aim to support transcription and decipherment of ciphertexts, developed to work together in a pipeline. It encompasses cataloging these documents into the Decode database, which houses ciphers dating from the 14th century to 1965, transcription using both manual and AI-assisted methods, cryptanalysis, and subsequent historical and linguistic analysis to contextualize decrypted content. The project encounters challenges with the accuracy of automated transcription technologies and the necessity for significant user involvement in the transcription and analysis processes. These insights highlight the critical balance between technological innovation and the indispensable input of domain expertise in advancing the field of historical cryptology.

1 Introduction

The Decrypt project (de-crypt.org) introduces a comprehensive suite of tools designed for the transcription and analysis of historical ciphers and keys. This initiative combines newly developed tools with enhanced versions of existing technologies to facilitate the study of historical encrypted sources. In this paper, we will give an outline of the tools, some under development, some in already in production — from transcription to decipherment — to work individually as well as together in a pipeline for efficient processing of historical encrypted sources. Noteworthy is that the pipeline is made accessible to users with an account to the decode database.

2 Elements of the Decrypt Pipeline

Our work centers on historical ciphers discovered across a diverse array of sources including both handwritten and printed documents of various lengths, featuring many distinct hand-writing styles and symbol sets. These documents, originating from different time periods and geographic regions in Europe, were created for a wide range of purposes such as diplomacy, military correspondence and even unconventional mediums like postcards.

The pipeline is illustrated in Figure 1 and will be described in the subsequent sections.

2.1 Cataloging Documents: the Decode Database

Upon discovery, historical ciphers or keys must be converted to digital images, accompanied by detailed metadata. This metadata typically includes details about the document's origins, like the sender and receiver, its creation or dispatch date and location as well as information about the medium. Accurately describing these historical sources with metadata is crucial as the contents are often completely unknown. This information can be recorded and stored in the Decode database (Héder and Megyesi, 2022). To make the processing of the encrypted sources easy, we developed an interface to allow users to create their own projects from which it is possible to start to utilize the entire pipeline for the analysis of the sources. Figure 2 shows a screenshot of user-specific projects.

2.2 Transcription

Once we have digital images, the next step is to turn the image into a machine readable text format, i.e. to provide a transcription of the content of the image. This step produces a document that contains the symbols seen on the digital images in a format that allows further processing, most often

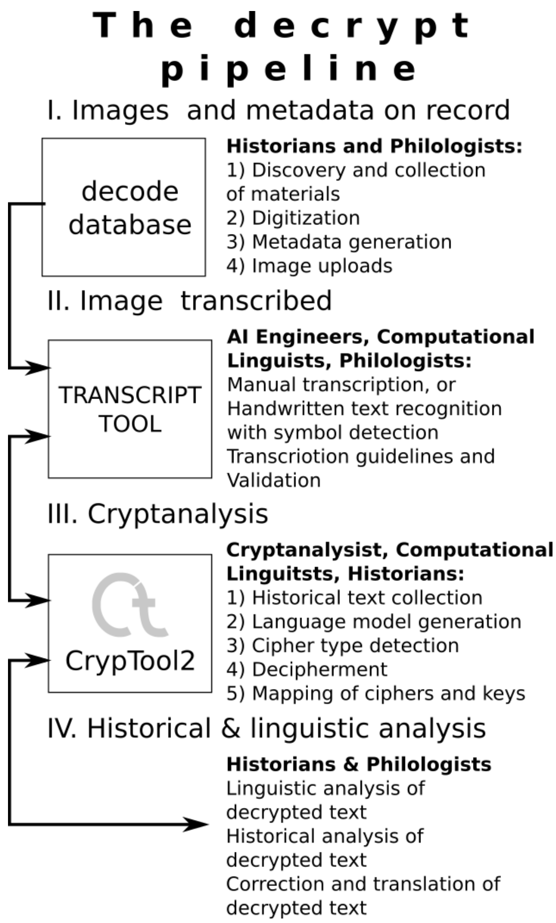

Figure 1: The Decrypt pipeline.

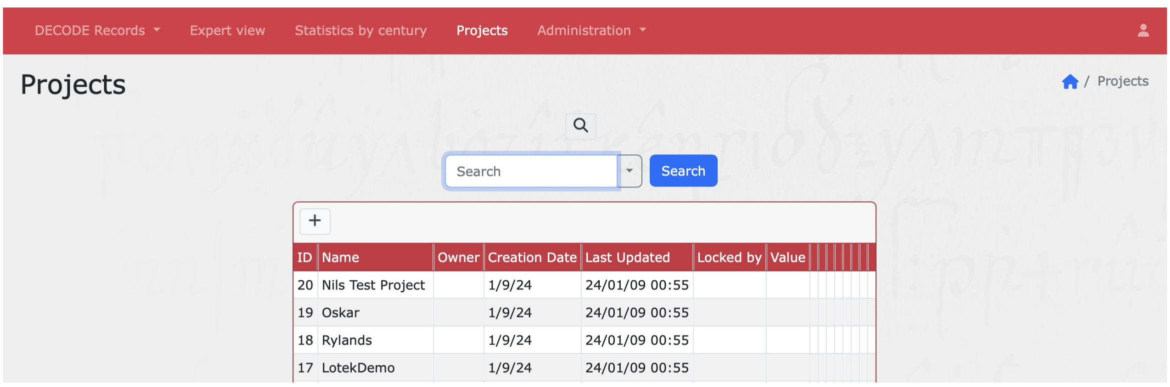

Figure 2: User-specific projects.

Unicode (UTF-8). The goal is to transcribe each symbol on the image as shown without leaving out any important information. Transcription can be carried out manually or (semi-)automatically.

George Lasry developed a tool for manual transcription called CrypTool Transcriber and Solver or shortly CTTS (Lasry et al., 2023), which was used, among others, for the transcription of the recently deciphered Mary Stuart letters. The tool is efficient but fully manual. CTTS allows the user to mark each symbol in the image and assign it into a cluster of similar symbols. Each cluster can then be assigned a label with which it can be transcribed in a unique way to result in a machine-readable text format. CTTS encourages a cyclic process of review and iteratively editing transcriptions and decryptions (Megyesi et al., 2024), which provides an excellent starting point to produce training data and manual transcriptions of shorter inscriptions that can be used to train better performing models. Figure 3 shows an image of CTTS currently being used to transcribe a ciphertext.

Transcription can be also done partly or entirely automatically using tools that are developed to specifically handle hand-written manuscripts. One of the most well-known transcription tools is Transkribus (Societas Cooperativa Europaea, 2024) which provides AI models trained on manually transcribed images for a wide range of European languages. However, these models are not directly applicable to encrypted sources, as these seldom contain character sequences known from plaintext languages. Therefore, the performance of these models are very low. We tested the three largest models of Transkribus on 20 different ciphertexts, all written as digits (0-9) and the models performed very poorly with a symbol error rate between 90% and 100%.

There is a great need of transcription tool(s) that can handle a wide range of symbol sets and handwriting styles. AI-based models can be an option, but they need large amounts of data, preferably transcribed ones to be used for training, which takes time to produce. Such AI systems requiring a lot of human interaction are usually referred to as human-in-the-loop AI. However, ciphertexts are usually short and training data is therefore often sparse. For this reason, some efforts have been put in developing tools that rely on efficient image processing algorithms and, at the same time, can learn from user feedback and corrections (Baró et al., 2019; Souibgui et al., 2021). Here, we turn our focus from a large set of manually produced training data to serve the AI to force the AI to learn from its own mistakes by correcting its output and fine-tune the model on the corrected data. We call our approach ”AI-in-the-loop” (Beáta Megyesi, 2022) since the AI assists the user instead of the other way around.

The TranscriptTool (Szigeti and Héder, 2022) was developed by the Decrypt project to support the user for fast and efficient transcription of cipher images. This tool is an AI-in-the-loop system which includes models trained on large set of training data, but it is also able to learn from user input. The system takes the output of a pre-trained image recognition algorithm and visualize it for the user for post-processing and error correction. The developed models are based on unsupervised clustering or few-shot learning, and they only require the manual correction of few lines. These models can be re-trained or fine-tuned with the examples provided by the user, which is of course a more resource-intensive process but leads to higher transcription performance. The TranscriptTool is illustrated in Figure 4 and described in more detail in (Megyesi et al., 2024).

CTTS and the TranscriptTool can be also used to mark the exact location of the symbols on the page (called ”bounding boxes”). This way the digital image of the individual symbols can be cut out to separate files. These files can serve further machine learning projects as real, historical instances of symbols.

2.3 Cryptanalysis

With a full or partial transcription at hand, the cryptanalysis may be attempted. This may involve statistical analysis, cipher symbol clustering (Lehofer, 2022), hill-climbing, simulated annealing, heuristics, or educated guesses. Historical language models for the suspected time period and language can be of great help, especially if the ciphertext originates from the 18th century or earlier (Megyesi et al., 2023). Cryptanalysis components for historical ciphers including simple and homophonic substitution of various complexity are integrated (Lasry et al., 2021) and available within CrypTool (CrypTool Contributors, 2024).

CrypTool has been developed as a desktop version, and now also as an online version.

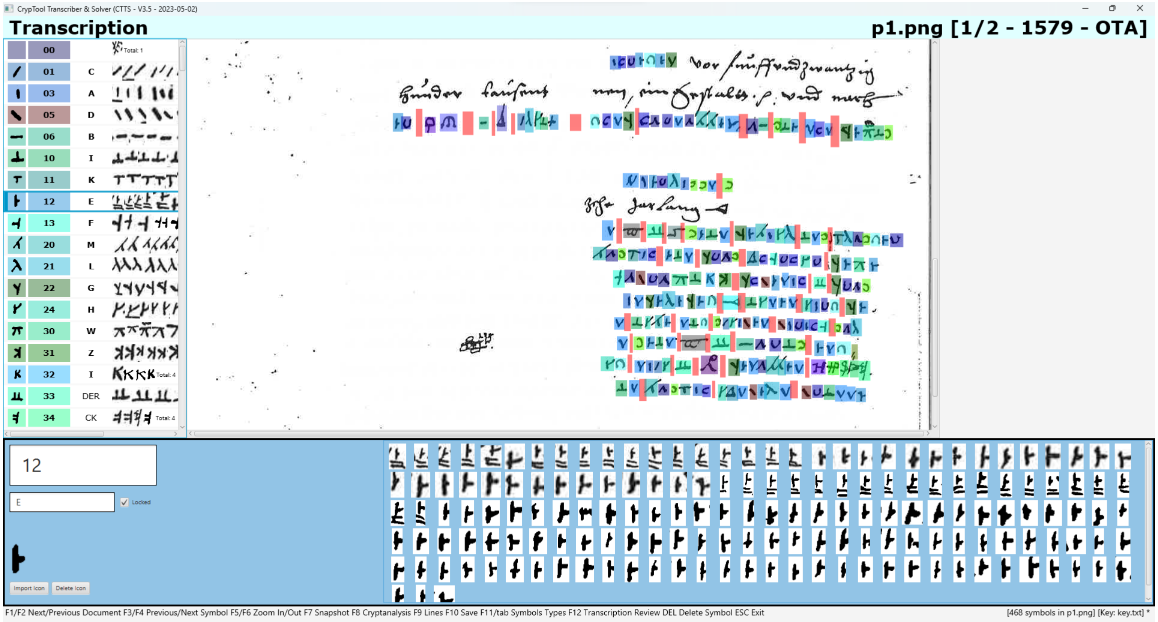

Figure 3: CrypTool Transcriber and Solver used to transcribe an image.

The desktop version, CrypTool 2 (Kopal, 2018), is aimed at more advanced users. It enables the creation of graphical programs, known as workspaces, which comprise various components for input, output, cryptography, and cryptanalysis working together to execute cryptographic or cryptanalytical tasks. A pre-created set of workspaces, referred to as templates, are included with CT2. These templates offer workflows for both cryptography and cryptanalysis, for instance, for cryptanalyzing simple substitution ciphers or homophonic substitution ciphers (Kopal, 2019). Additionally, templates designed for conducting statistical analyses (e.g., frequency analysis, Friedman test) on ciphertexts are part of the software. Currently, CT2 boasts over 250 such templates. It has been successfully utilized in cryptanalyzing different ciphertexts, such as written letters from Maximilian II from 1575 (Kopal and Waldispühl, 2022) or an encrypted letter from the Dutch East India Company (Dinnissen and Kopal, 2021). We are currently extending CT2 with the so-called DECRYPT editor, a specialized component of CT2 that facilitates easy access to the DECRYPT pipeline and provides a comprehensive set of cryptanalysis tools at a single point. Figure 5 illustrates the project overview on the left side and a selected project on the right. On the lower right side, the files associated with the project are displayed. Figure 6 depicts a frequency analysis of a ciphertext conducted using the DECRYPT editor. This tool allows for the flexible selection of the ciphertext's alphabet, enabling, for example, the analysis of digit distribution in digit ciphers. The CT2 DECRYPT editor is currently under development and will be published in the near future.

In addition to CT2, there is an online counterpart, CrypTool-Online (CTO). This web-based version does not require any installation, distinguishing it from CT2. CTO features a variety of cryptanalysis tools, notably including the new online homophonic substitution analyzer. This tool enables users to manually analyze homophonic substitution ciphers (automatic cryptanalysis is also planned) by assigning plaintext letters to ciphertext symbols. It highlights all instances of a selected symbol and automatically assigns the chosen plaintext letter to all occurrences of that symbol. A key advantage of the online version over CT2's analyzer is its support for the DECRYPT transcription guidelines (Megyesi, 2020). It accommodates not only single-letter homophones but also vowels and complex n-grams. Furthermore, the tool facilitates the annotation of nomenclature elements and inline plaintext elements, adhering to the DECRYPT guidelines. Figure 7 displays a ciphertext under analysis with the Homophonic Substitution Analyzer in CTO. Currently in beta, the online version is set to be launched in the coming months.

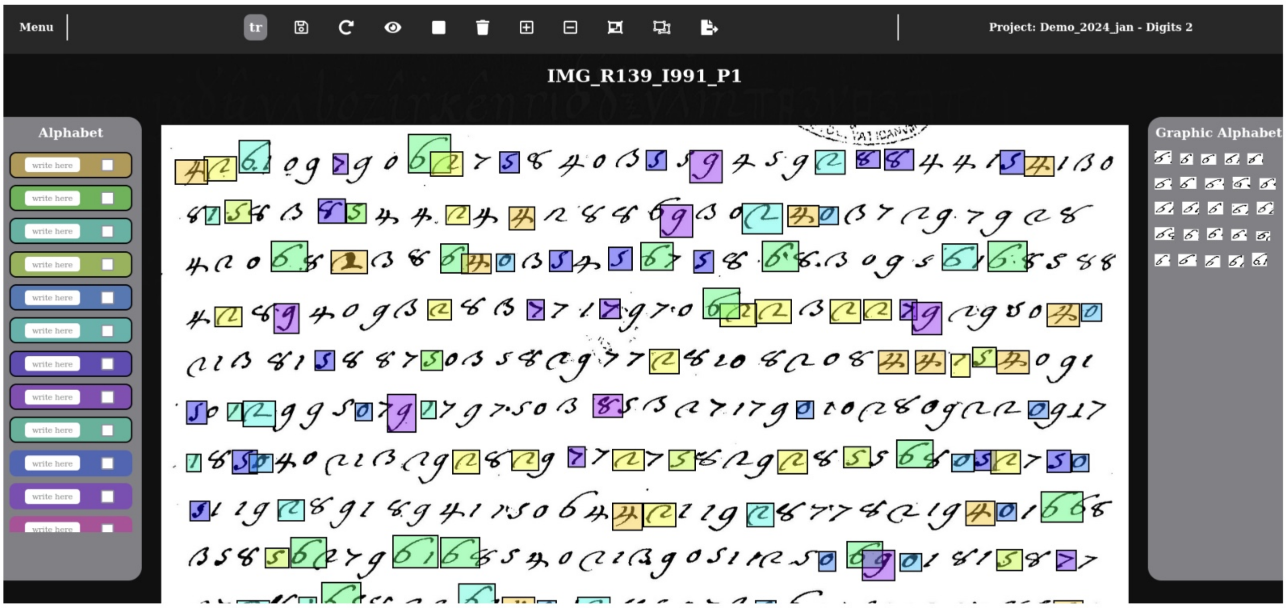

Figure 4: The TranscriptTool.

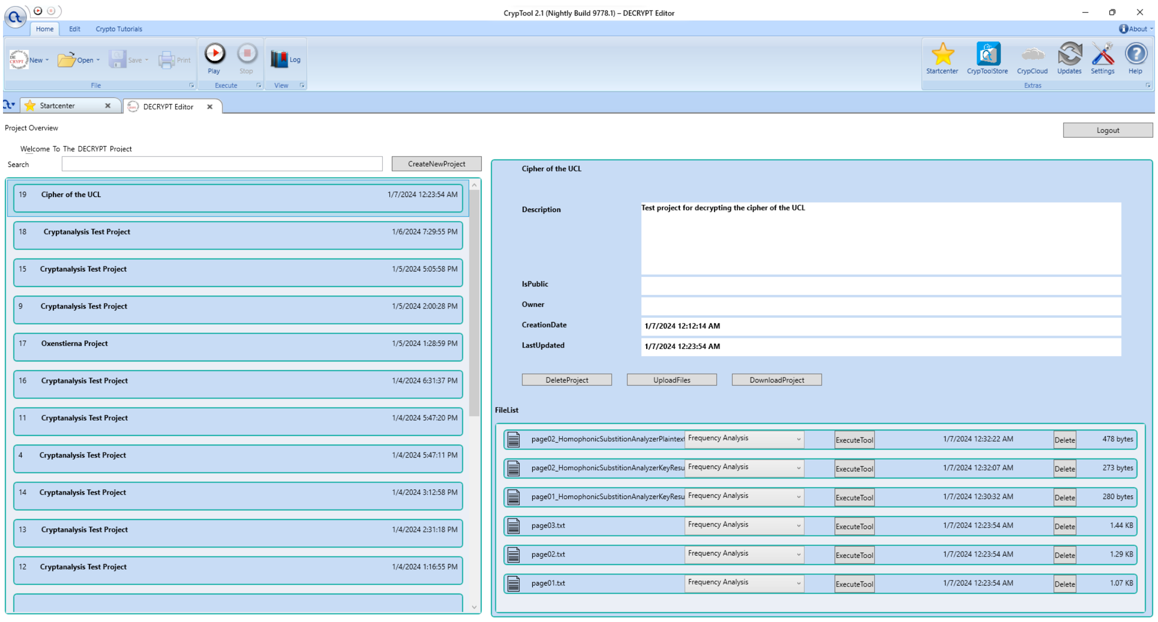

Figure 5: DECRYPT editor in CrypTool 2.

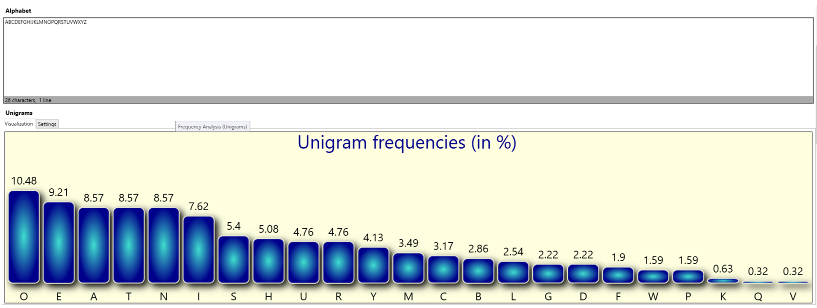

Figure 6: Frequency analysis conducted using the DECRYPT editor, which allows for the use of a user-defined alphabet (in this case, Latin), within CrypTool 2.

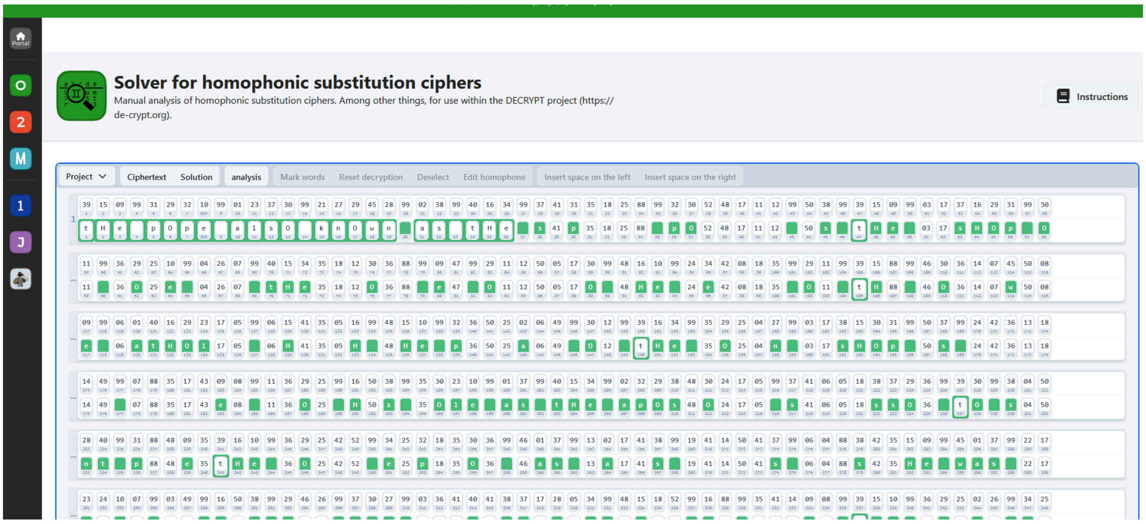

Figure 7: Cryptanalysis performed with CrypTool-Online's Homophonic Substitution Analyzer.

2.4 Historical and Linguistic Analysis

If the decryption is successful, the historian and the philologist may draw further conclusions. Usually, there is a correction and translation is necessary to make the content intelligible — sometimes even within the same language, i.e. from old Hungarian to the present day version. As the content, even in deciphered form, may lack context like sender, receiver or dates and locations, there is further work to be performed by the historian. The updated information, whether it is metadata or additional documents, can be uploaded to further enrich the Decode database. The upload can be made directly from the user-specific project or in connection to the record in the database.

3 Limitations and Lessons Learned

One main limitation of this pipeline, after evaluation by the Decrypt team, is the error rate of symbol recognition with image processing. Even with the best methods, the character error rate remains too high in order for the user to rely on the system without systematic verification error correction. This, in turn, makes document pre-processing and transcription similarly resource-intensive as manual processing, at least in case of shorter documents.

The Decrypt team also realized that while the manuscript transcription and cryptanalysis may be automatized to a large degree, the user's domain knowledge still needs to be present and the learning curve to that knowledge can be quite high.

3.1 The Connecting Fabric: API

Every action that can be conducted on the web interface of the DECODE database including the DECRYPT project can be managed via an application programming interface (API). This allows that elements of the pipeline store intermediate results, configurations and other information on the server. Moreover, the API also has been used to gather aggregated statistical data from about the several thousand records in the DECODE database. Figure 8 shows the auto-generated documentation and sandbox interface of the API.

4 Conclusion

The Decrypt project's pipeline represents a groundbreaking endeavor in the field of historical cryptology, offering a multi-faceted approach to the analysis of historical ciphers. Despite facing challenges related to the accuracy of automated transcription and the necessity for user involvement, the project illustrates the potential for technology to significantly aid in deciphering historical documents. The lessons learned underscore the importance of combining technological innovation with expert knowledge, paving the way for future advancements in the study of historical cryptology.

Acknowledgments

This work has been supported by the Swedish Research Council, grant 2018-06074, DECRYPT – Decryption of Historical Manuscripts.

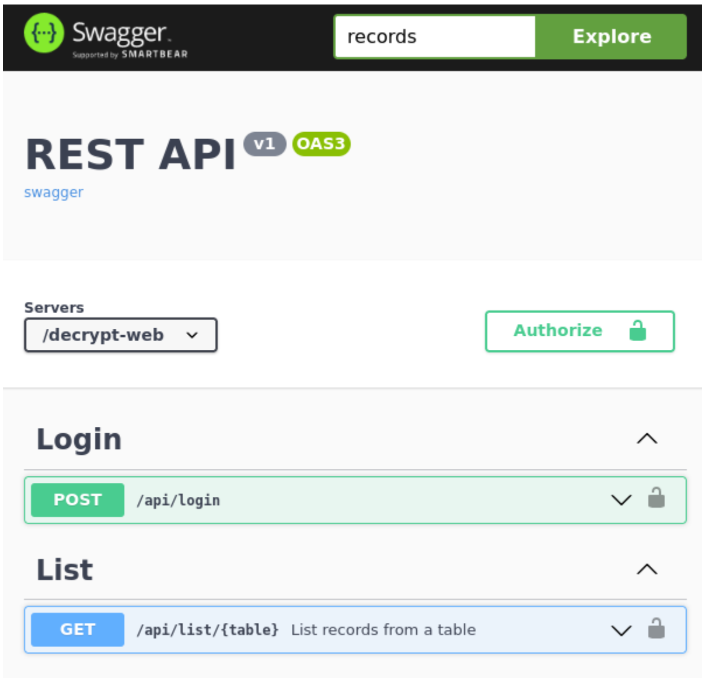

Figure 8: API.

References

Arnau Baró, Jialuo Chen, Alicia Fornés, and Beáta Megyesi. 2019. Towards a generic unsupervised method for transcription of encoded manuscripts. In *3rd International Conference on Digital Access to Textual Cultural Heritage (DATECH)*, pages 73–78.

Beáta Megyesi. 2022. Cracking Historical Ciphers with AI in the Loop. Invited talk at Colby College, USA May 6, 2022.

CrypTool Contributors. 2024. Cryptool 2 - cryptool portal.

Jörgen Dinnissen and Nils Kopal. 2021. Island Ramanacoil a Bridge too Far. A Dutch Ciphertext from 1674. In *Proceedings of the 4th International Conference on Historical Cryptology, HistoCrypt 2021*, pages 48–57.

Mihály Héder and Beáta Megyesi. 2022. The DECODE Database of Historical Ciphers and Keys: Version 2. In *Proceedings of the 5th International Conference on Historical Cryptology, HistoCrypt22*.

Nils Kopal and Michelle Waldispühl. 2022. Deciphering three diplomatic letters sent by Maximilian II in 1575. *Cryptologia*, 46(2):103–127.

Nils Kopal. 2018. Solving Classical Ciphers with CrypTool 2. In *Proceedings of the 1st International Conference on Historical Cryptology HistoCrypt 2018*, number 149, pages 29–38.

Nils Kopal. 2019. Cryptanalysis of Homophonic Substitution Ciphers using Simulated Annealing with Fixed Temperature. In *Proceedings of the 2nd International Conference on Historical Cryptology, HistoCrypt*, pages 107–16.

George Lasry, Beáta Megyesi, and Nils Kopal. 2021. Deciphering Papal Ciphers from the 16th to the 18th Century. *Cryptologia*, 45(6):479–540.

George Lasry, Norbert Biermann, and Satoshi Tomokiyo. 2023. Deciphering mary stuart's lost letters from 1578-1584. *Cryptologia*, 47(2):101–202.

Anna Lehofer. 2022. Applying hierarchical clustering to homophonic substitution ciphers using historical corpora. *Cryptologia*, 46(5):422–438.

Beáta Megyesi, Justyna Sikora, Filip Fornmark, Michelle Waldispühl, Nils Kopal, and Vasily Mikhalev. 2023. Historical Language Models in Cryptanalysis: Case Studies on English and German. In *International Conference on Historical Cryptology*, pages 120–129.

Beáta Megyesi, Alicia Fornés, Nils Kopal, Benedek Láng, Michelle Waldispühl, Vasily Mikhalev, and Bernhard Esslinger. 2024. Historical cryptology. *In Ed: Bernhard Esslinger Learning and Experiencing Cryptography with CrypTool and Sagemath*.

Beáta Megyesi. 2020. Transcription of Historical Ciphers and Keys. In *Proceedings of the 3rd International Conference on Historical Cryptology, HistoCrypt20*, Budapest, Hungary, June.

Societas Cooperativa Europaea. 2024. Transkribus.ai.

Mohamed Ali Souibgui, Alicia Fornés, Yousri Kessentini, and Beáta Megyesi. 2021. Few shots are all you need: A progressive few shot learning approach for low resource handwritten text recognition. *arXiv preprint arXiv:2107.10064*.

Ferenc Szigeti and Mihály Héder. 2022. The TRANSCRIPT tool for historical ciphers by the DECRYPT project. In *Proceedings of the 5th International Conference on Historical Cryptology*, pages 208–211.