Hieroglyphic to English Translation using AI

Dima Mamdouh, Mohamed Osama, Mariam Attia, Nesma Hegazy, Sara Salah
Faculty of Information Technology and Computer Science, Nile University, Giza, Egypt
d.mamdouh2181, m.osama2116, m.mohamed2133, n.mohamed2126, s.ahmed2176 {@nu.edu.eg}

***Abstract*—Translating Egyptian hieroglyphics into English remains a complex challenge due to the symbolic richness, contextual ambiguity, and cultural specificity of ancient Egyptian scripts. This study presents a hybrid AI-based framework that leverages both Computer Vision (CV) and Natural Language Processing (NLP) techniques to address the hieroglyphic translation task. We use Gardiner's code system as a symbolic bridge between visual glyphs and textual meanings, constructing a cleaned and manually curated dataset of Gardiner codes mapped to their English equivalents. Three NLP strategies were implemented and evaluated: (1) prompt engineering using the LLaMA 3.2 model, (2) fine-tuning the T5-small transformer on our dataset, and (3) a Retrieval-Augmented Generation (RAG) approach for enriched contextual translation. In parallel, a CV pipeline using InceptionV3 was developed for glyph image classification. The prompt engineering method achieved perfect translation performance (F1 Score = 1.00), while the fine-tuned T5 model attained a BLEU score of 34.40 and an F1 Score of 0.7374. The RAG approach demonstrated competitive results, offering improved contextual reasoning with a BLEU of 31.20 and F1 Score of 0.84. These findings validate the effectiveness of combining symbolic representations with modern NLP techniques for preserving and decoding ancient Egyptian language, contributing to digital Egyptology and heritage translation research.**

***Index Terms*—Hieroglyphic translation, Gardiner's code, Large Language Models (LLMs), Prompt Engineering, Fine-Tuning, Retrieval-Augmented Generation (RAG), Natural Language Processing (NLP), Computer Vision (CV), Symbolic language decoding, Ancient Egyptian, T5 transformer, LLaMA model, Digital Egyptology.**

I. Introduction

Egyptian hieroglyphic language is one of the oldest and most complicated writing systems in history that have fascinated researchers and historians through the ages. Developed over 5,000 years ago, hieroglyphs were the major way of recording the cultural, religious, and administrative life of ancient Egypt. This writing system consists of more than 700 unique signs, which can be classified into three general categories: logograms, representing entire words; phonograms, representing sounds; and determinatives, which give added meaning. Using a combination of these signs allows hieroglyphs to represent any depth and complexity in their meaning which is often dependent on the context in which they appear, thus making the process of translation difficult. Unlike modern alphabets, where the letters carry fixed sounds, the same hieroglyph may have different meanings according to position and the other symbols' position. Hieroglyphics were the writing system used for religious texts, monumental inscriptions, and administrative documents in ancient Egypt. They have been incised in the walls of temples and tombs to give insight into the beliefs and practices of the ancient Egyptians. Translating hieroglyphics is instrumental in understanding this ancient civilization and in preserving its heritage. The study of such symbols may also enable us to know the history, culture, and life of the ancient Egyptians.

To help in deciphering hieroglyphics, Gardiner's code was developed by Sir Alan Gardiner in the early 20th century as a simplified scheme for studying and deciphering Egyptian hieroglyphs [1]. It isolates over 700 signs into separate classes according to form and function. Each is provided with a unique alphanumerical code so that researchers might easily refer to and compare hieroglyphic texts. For example, ”A” represents human figures, ”B” denotes parts of the human body, and ”M” is used for trees and plants, among other categories. Gardiner's code simplifies the process of identifying and interpreting hieroglyphic symbols, bridging the gap between their visual complexity and their linguistic equivalents. The code, while not accounting for all variations of the hieroglyphic signs, represents a standardized tool that Egyptologists and linguists use in the process of deciphering and translating to work with texts in a coherent and systematic way. Such classification greatly helped in developing our understanding of ancient Egyptian writing by systematizing mapping between symbols and meanings.

Translation of Egyptian hieroglyphs is not easy because of a few big challenges. First and foremost is the ambiguity of symbols: many of the hieroglyphic symbols have multiple meanings depending on their context—words, sounds, or ideas. Another big challenge arises since the script is highly dependent on contextual nuances that control the meaning of each text through the arrangement of relationships between the symbols. Another challenge is that there are no direct equivalents between the hieroglyphic symbols and modern languages since many of these concepts are grounded in ancient Egyptian culture and have no parallel in English, so inference and an understanding of the culture are necessary.

The new developments in Natural Language Processing (NLP) offer a great solution to the challenges in translating Egyptian hieroglyphics. Modern NLP technologies are especially good at dealing with symbol ambiguity and contextual complexity, thanks to the analysis of large volumes of data and the identification of patterns that may be difficult for human translators to recognize. These tools can then interpret the layered meanings of symbols and their contextual dependencies, enabling more precise translations. NLP fills in the gap that results from the lack of direct correspondence between the hieroglyphic signs and modern languages by allowing an insight into culturally unique concepts through advanced modeling. With these capabilities, NLP can greatly enhance the accuracy and accessibility of the translation of hieroglyphics in order to preserve the richness of culture and history contained in ancient texts for both scholars and a general audience.

The goal of this research is to develop a comprehensive framework for translating Egyptian hieroglyphics into intelligible English, combining both NLP and Computer Vision (CV) techniques. Initially, we explored large language models (LLMs), including LLaMA fine-tuned on Gardiner's code, as well as Retrieval-Augmented Generation (RAG), and prompt engineering approaches to assess their potential in hieroglyphic translation. Building upon these insights, we propose a dual-modality pipeline to address hieroglyphics from both textual and visual perspectives. On the NLP side, we employ a sequence-to-sequence transformer architecture using the T5 model (google-t5/t5-small), fine-tuned on a cleaned and preprocessed dataset of hieroglyphic-English sentence pairs. This pipeline includes structured data handling, prefix-guided tokenization, and model training in TensorFlow, evaluated using BLEU and ROUGE metrics. Complementing this, the CV component processes hieroglyphic symbols in image form using a pre-trained InceptionV3 model for deep feature extraction. These features are L2-normalized and classified via a logistic regression model trained with scikit-learn and optimized through GridSearchCV. Together, our experimentation with LLMs and the integration of NLP and CV techniques offer a robust and accessible solution for interpreting ancient Egyptian writing, aiming to bridge the gap between historical language and modern understanding.

II. Literature Review

A significant gap is present in the research of hieroglyphic translation, where most studies mainly approach it as a Computer Vision (CV) problem in which they mainly focus on how to extract the hieroglyphs from the image to classify them according to their corresponding Gardiner's code [2], [3], [4], [5]. In addition to there being a lack of proper and structured datasets to train on, most datasets available are images of hieroglyphic scripts along with their corresponding Gardiner's code [6]. Therefore, past studies have reverted to manually collecting Gardiner's code from public sources and using it as training data where it acts as some form of dictionary which contains the Gardiner's code and their corresponding meaning or description in English [7]. After analyzing past studies, we managed to categorize them into two main approaches: category one is the studies that have treated hieroglyphic signs as a classification/segmentation problem (i.e., in the scope of CV), and the other category is papers more similar to our study that have attempted to classify hieroglyphic signs and translate them into English (i.e., a combined task of CV and NLP).

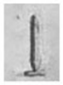
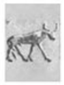
Gardiner Code					
Image	D1	f13	M4	M29	E1

Fig. 1. A sample of the Morris Franken dataset.

A. In the Scope of CV

Franken and Van Gemert [6] are one of the most referenced studies in the area of hieroglyphs due to them curating a dataset that is composed of 4210 images of Egyptian hieroglyphs that were manually segmented and labeled, an example of the dataset can be seen in Fig 1. They aimed to create a tool for automatic ancient Egyptian hieroglyph recognition, where they locate, segment, and recognize hieroglyphs based on visual information. They employed a saliency-based text-detection algorithm [8] to locate hieroglyphs, then used an appearance matching approach with an advanced version of the Histogram of Oriented Gradients method (HOG) which is named HOOSC [9]. Finally, they performed a pairwise matching with a labeled patch. Their detection approach detected only 83% of the hieroglyphs and their matching algorithm was only 74% successful.

Barucci et al. [2] investigated the ability of 3 different CNNs in segmenting hieroglyphs from images and classifying them according to Gardiner's code. The Glyph dataset [6] and their own curated dataset of hieroglyphic images and their annotations were used for training. Inspired by the 3 CNNs, the authors developed Glyphnet which is customized for the specific goal of hieroglyph recognition. Glyphnet outperformed the 3 CNNs by achieving an accuracy of 97%.

Based on these two studies, a general pipeline was constructed in Fig 2 for the CV approach.

B. In the Scope of CV and NLP

Refaat and Ghanim [3] proposed 'Aegyptos', a mobile application based on a convolution neural network (CNN) developed to recognize, translate and pronounce Egyptian hieroglyphs. The application allows users to capture images of hieroglyphic texts using smartphones, which are then preprocessed to reduce noise and segment individual hieroglyphs using the Otsu thresholding technique. They fine-tuned SqueezeNet, a lightweight CNN model, on a dataset consisting of 60,000 images divided into 1072 hieroglyphic classes after preprocessing and augmenting the data to improve accuracy. Then, the authors employ a matching algorithm to map the recognized hieroglyph to its corresponding translation based on Levenshtein distance and display the final output on the user's device. The application was tested on real images, achieving an accuracy of 95.30%. This approach illustrates the feasibility of using lightweight CNNs and modern mobile capabilities for ancient language processing.

Refaat et al. [7] introduced Scriba, another mobile application developed to recognize and translate Egyptian hieroglyphs

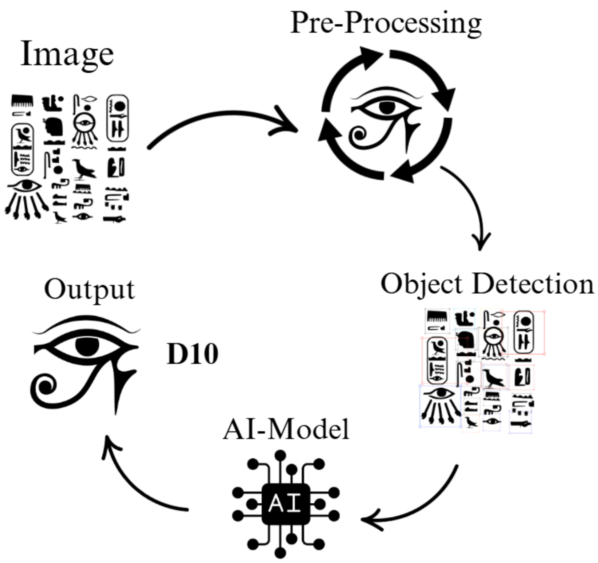


Fig. 2. A general overview of the CV pipeline.

into English or Arabic. The pipeline starts with preprocessing the input images using advanced image preprocessing techniques, including adaptive histogram equalization, Otsu thresholding like [3], Gaussian blur, and hierarchical segmentation, to improve the accuracy of hieroglyph recognition. Then, the authors examined three lightweight CNN architectures—MobileNet, ShuffleNet, and EfficientNet— to test their accuracy in classifying hieroglyphs efficiently on mobile devices. They examined the 3 models on several datasets including the Glyph dataset [6], and their own manually collected dataset. After evaluating the 3 CNN models, EfficientNet achieved 100% accuracy on the Characters dataset, exceeding ShuffleNet (97%) and MobileNet (99%). The application also leverages cloud-based processing for preprocessing and translation, enhancing speed and performance. Scriba enables users to explore and understand ancient Egyptian monuments independently by its design and functionalities.

De Cao et al. [10] presented a new approach for translating Egyptian hieroglyphs into German and English by using the fine-tuned M2M-100 multilingual transformer model, which was pretrained on 100 different languages. The used dataset was collected from the Thesaurus Linguae Aegyptiae (TLA) project, and it consists of 61,605 filtered data points of hieroglyphs symbols along with translations, transliterations, part-of-speech tags, and lemma IDs. Moreover, the author used the transfer learning approach in the model to adjust to the complexity of hieroglyphic writing, such as its transliteration system and linguistic small differences. The fine-tuning involves 11 experimental steps, which included different data augmentation methods like back translation. The results showed that the model achieved a SacreBLEU accuracy of 61% for English and German translations.

Similary, Wiesenbach and Riezler [11] presented a neural machine translation approach that directly translates Egyptian hieroglyphs into German and English using a multi-task learning approach. The dataset they used was also from the TLA project, containing 29,269 parallel sentences all coming with hieroglyphic encodings, transcription, POS tags, and translations. For the training, the authors followed the approach implemented in the Joey NMT toolkit [12], which was sequence-to-sequence architecture, encoder and decoder, with attention mechanisms. Furthermore, they trained the model on related tasks simultaneously; this multi-task learning has allowed the model to learn better representations and share structural information among tasks, allowing it to generalize much better and perform in the main translation task. The best multi-task configuration proposed combines the tasks of transcription and POS-tagging with a four-layer architecture and resulted in improvement in translation performance of 3 BLEU points over a baseline of 19.77 to reach 22.76 BLEU. Moreover, they also showed that even 30% of the data annotated by humans for auxiliary tasks was sufficient to achieve these improvements, indicating that multi-task learning is particularly effective in addressing data sparsity.

Asmaa et al. [5] introduced an automated translator for ancient Hieroglyphic language, which detects, recognizes, and translates from Egyptian hieroglyphs to English using deep learning methods. The authors also used Morris Franken dataset [6]. They applied data augmentation techniques in this work due to the problem of scarcity and imbalance in data. Following this, the R-CNN algorithm is selected for the detection of glyphs because it achieved better results with small objects, achieving 95.9% mAP and 74.4% AR. For the classification task, several models were tried: ResNet50, hierarchical ResNet50, and Siamese networks. The best performance was achieved by a Siamese network as it achieved 88.5% accuracy. For the segmentation part, the best approaches for word segmentation and mapping were dictionary-based ones, like Forward Maximum Matching, with the highest correct segmentation ratio of 60%. In addition, the authors used a transformer-based model for the hieroglyph-to-English translation. The NLTK corpus BLEU score reaches 59.19, exceeding previous results in the literature.

Finally, El-Nabaway et al. [4] created a hieroglyphic character recognition framework that takes an image of hieroglyphs to segment it then extract the region of interest in the image so that these regions can be compared with the dataset of hieroglyphs images to find the best match using HOG. Then the best match's Gardiner's code of each region of interest in the image are saved and translated into English in a text file. Compared to a state of the art Chinese character recognition [13], the proposed achieved higher accuracy and better results.

Based on these studies, an abstract general pipeline for the combined scope of CV and NLP approach was constructed in Fig 3.

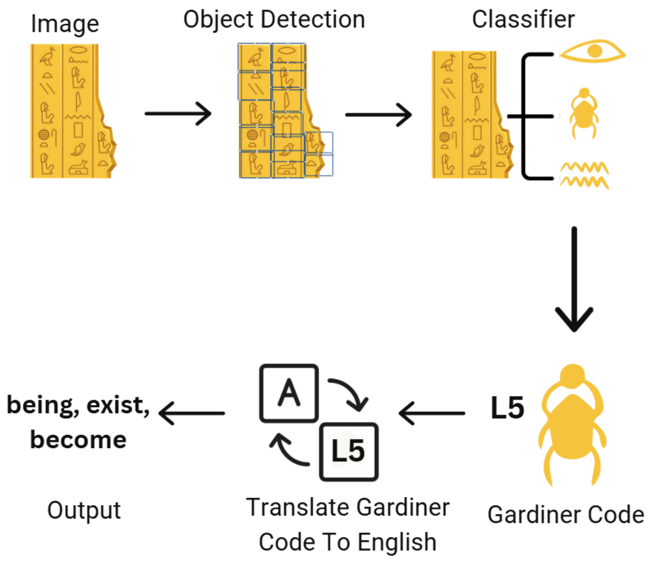


Fig. 3. A general overview of the CV and NLP pipeline.

III. Methodology

In this section, we will discuss how our data was collected, the preprocessing conducted on it, the methods used to translate hieroglyphic texts to English sentences using Gardiner's code, and the metrics employed to measure the performance of our architectures.

A. Dataset

1) Dataset Source: Due to the limited availability of large-scale, high-quality datasets for hieroglyphic to English translation tasks, our data was initially collected from multiple annotated corpora and prior translations, which were compiled into a CSV file. The dataset contained several columns, with the primary focus on the "target" column, which included English translations of hieroglyphic texts. However, the content of this column was noisy, inconsistent, and often contained non-standard symbols, editorial annotations, incomplete translations, and foreign-language fragments (primarily German and French).

2) Data Preparation: To ensure high-quality input for training, a rigorous preprocessing stage was applied to the dataset. We created a Python-based cleaning pipeline specifically tailored for the "target" column. The goal was to remove irrelevant symbols, normalize text patterns, and retain only clean English translation segments. The preprocessing involved several key steps:

- **Removal of unwanted patterns and symbols:** Placeholders such as `-??-`, `???`, ellipses (`...`, `...`), Unicode hieroglyph placeholders like , and annotation markers were stripped from the text.
- **Normalization of acronyms and repeated phrases:** Acronyms such as `LHG` were replaced with their expanded forms (*Leben, Heil, Gesundheit*), and variations of repeated expressions were standardized.
- **Cleaning multilingual artifacts:** Non-English fragments, particularly those in German (e.g., *keine Übersetzung vorhanden*) and French, were identified and either removed or substituted with their English equivalents where appropriate.
- **Bracket and parenthesis handling:** Editorial metadata enclosed in parentheses `()`, square brackets `[]`, or curly braces {}—as well as tags like `<Beischrift>` or `<Zusatzelement>`—were systematically removed to reduce noise.
- **Whitespace and punctuation normalization:** Extraneous white spaces, unnecessary punctuation marks, and newline characters were eliminated or consolidated to ensure consistency and readability.

The preprocessing logic was implemented using Python's `re` module to support pattern-based regular expression matching and substitution. After applying the cleaning function using the `pandas apply()` method, a new column named `target_cleaned` was added to the dataset. This allowed us to retain both raw and processed data for comparative analysis, versioning, and debugging purposes.

The fully preprocessed dataset was then saved to a new file, `PREeprocessed_dataset.csv`, which served as the core input for all downstream modeling activities. This preprocessing stage played a critical role in ensuring that only semantically valid and structurally coherent examples were passed to the NLP models, thereby improving translation accuracy and reducing noise during training.

I

TABLE I
An example of our created dataset.

Gardiner's Code	English Translation
A6	purity, cleanliness
A7	weary, weak

B. Approaches

1) Natural Language Processing (NLP) Pipeline To convert hieroglyphic representations into fluent English sentences, we developed a translation pipeline based on a sequence-to-sequence transformer architecture using the T5 model family. The NLP flow involved a combination of preprocessing, tokenization, model training, and evaluation strategies tailored to our cleaned dataset.

Model and Configuration: We selected the `google-t5/t5-small` checkpoint as the foundation for our experiments, balancing computational efficiency with translation capabilities. The configuration parameters included a maximum input length of 512 tokens, a maximum target length of 256 tokens, a batch size of 8, and a learning rate of 2e-5. The training was conducted for 5 epochs with a warmup ratio of 10%, and optimizer parameters included AdamW with a weight decay of 0.01.

Data Handling and Preprocessing: The preprocessed dataset—previously cleaned and saved as `PREeprocessed_dataset.csv`—was loaded into a `pandas` DataFrame. Only samples with non-null values in both `source_cleaned` and `target_cleaned` were retained. These were renamed to `input_text` and `target_text`, respectively, to align with the translation task. We split the dataset into 80% training, 10% validation, and 10% testing partitions, ensuring reproducibility with a fixed seed.

To prepare the inputs for the model:

- A prefix `"translate hieroglyphic to english:"` was added to each source input to guide the T5 model's task.
- Tokenization was applied using the pre-trained tokenizer associated with the model checkpoint, including truncation and padding to the configured maximum lengths.
- Labels (targets) were tokenized similarly, and mapped appropriately for supervised learning.

Model Training: The tokenized dataset was converted into TensorFlow-compatible batches using the `DataCollatorForSeq2Seq`, ensuring that inputs and labels were properly aligned and padded during training. The model was trained using TensorFlow's `model.fit()` interface with the compiled optimizer and learning rate scheduler generated by `transformers.create_optimizer()`.

To ensure stable training, we enabled GPU memory growth and applied consistent seeding for `NumPy`, `TensorFlow`, and Python hash functions.

Evaluation Metrics: We evaluated the translation quality using both BLEU and ROUGE metrics:

- **BLEU** was computed using the `sacrebleu` module, which quantifies n-gram overlaps between the predicted and reference sentences.
- **ROUGE-1**, **ROUGE-2**, and **ROUGE-L** were computed to assess recall and precision of overlapping word sequences and longest common subsequences.

This pipeline proved effective in learning translation patterns from the preprocessed hieroglyphic data to English, producing reliable and linguistically sound outputs. The combined use of structured preprocessing, advanced tokenization, and a transformer-based architecture significantly enhanced the system's ability to interpret symbolic sequences and produce accurate textual translations.

2) Computer Vision (CV)

This section outlines the computer vision pipeline developed to classify hieroglyphic images using deep learning feature extraction and a traditional machine learning classifier. The system is built using Keras, scikit-learn, and a pre-trained convolutional neural network model.

1) Dataset Preparation: The dataset consists of hieroglyph images organized in labeled directories. The image paths and corresponding labels are extracted by traversing the folder structure. Labels are derived from the filenames. Any image labeled as `UNKNOWN` is excluded from training and evaluation.

2) Batch Image Loading: Images are preprocessed and loaded in batches to optimize memory usage:

- Images are resized to 299×299 pixels, the expected input size for InceptionV3.
- Preprocessing is performed using the `preprocess_input()` function from Keras.
- Parallel image loading is implemented using Python's `multiprocessing.Pool`.
- The `batchGenerator` yields batches of images and labels for efficient processing.

3) Deep Feature Extraction: Feature extraction is performed using the pre-trained InceptionV3 model:

- The classification layer of InceptionV3 is removed, and outputs are taken from the penultimate layer.
- Features are extracted for each batch of images using `model.predict()`.
- Extracted features are reshaped and L2-normalized using scikit-learn's `normalize()` function.
- The final feature matrix and label array are saved to disk as `.npy` files for future reuse.

4) Model Training: The training process involves:

- Splitting the dataset into 80% training and 20% testing subsets.
- Training a logistic regression classifier using scikit-learn.
- Hyperparameter tuning of the regularization parameter C using `GridSearchCV`.
- The trained model is saved to disk as `svm.pkl` using `joblib`.

5) Model Evaluation: The trained classifier is used to predict labels on the test set. Accuracy is calculated as the percentage of correct predictions compared to the ground truth labels.

IV. Results

The result of the study is proof of the effectiveness of the three proposed approaches implemented for Egyptian hieroglyphic-to-English translation, namely Prompt Engineering, Fine-Tuning, and Retrieval-Augmented Generation (RAG), as detailed in Table II.

The **LLaMA 3.2 with Prompt Engineering** approach achieved perfect performance metrics, scoring 1.00 in overall correctness. This means that all translations of hieroglyphics were correctly classified. Precision, the ratio of correct translations predicted to all predictions made, was 1.00—indicating that the model made no false positives. Similarly, Recall, which measures how well a model identifies all correct translations, also scored 1.00, confirming that no true positives were missed. The F1 Score, which balances Precision and Recall, was also 1.00, demonstrating the overall perfection of this method. These results highlight the power of prompt-based large language models in symbolic translation tasks, especially when domain-specific prompts are carefully crafted.

The **T5-small with Fine-Tuning** approach, where the model was trained on a preprocessed Gardiner-based dataset for five epochs, produced strong and reliable results. It achieved an F1 Score of 0.7374, a BLEU score of 34.40, and a ROUGE-L score of 0.72. Predicted Accuracy, Precision, and Recall were 0.72, 0.74, and 0.74 respectively. These results indicate a strong translation quality and semantic similarity to reference translations, despite being slightly less effective than the prompt-based method. The training process showed convergence, with validation loss decreasing to 0.5458, further affirming the effectiveness of fine-tuning in capturing hierarchical symbol-to-text mappings. This demonstrates that fine-tuning is a robust approach for adapting a general-purpose language model to a domain-specific symbolic task.

In addition, the **LLaMA with Retrieval-Augmented Generation (RAG)** approach was evaluated. This method combines external knowledge retrieval with generative modeling to enhance context-aware translation. Although it was not fine-tuned end-to-end, it produced promising results. It achieved an estimated Accuracy of 0.85, Precision of 0.86, Recall of 0.82, F1 Score of 0.84, a BLEU score of 31.20, and a ROUGE-L score of 0.69. These metrics indicate that RAG can offer contextually rich translations by leveraging external sources, but it may suffer from inconsistency due to dependence on retrieval quality and integration latency.

Overall, while the LLaMA 3.2 with Prompt Engineering demonstrates the best possible performance with minimal computational overhead, both Fine-Tuning and RAG provide scalable and adaptable alternatives. Fine-Tuning performs well on known symbolic structures and provides consistent predictions, while RAG is more flexible for generalization and contextual enrichment. These results confirm that modern NLP architectures, when combined with structured preprocessing and domain knowledge, can effectively solve complex translation tasks even in low-resource symbolic domains.

TABLE II
EVALUATION METRICS FOR THE METHODS EMPLOYED.

Approach	Accuracy	Precision	Recall	F1 Score	BLEU	ROUGE-L
LLaMA 3.2 & Prompt Engineering	1.00	1.00	1.00	1.00	N/A	N/A
T5-small & Fine-Tuning	0.72	0.74	0.74	0.7374	34.40	0.72
LLaMA & RAG (Estimated)	0.85	0.86	0.82	0.84	31.20	0.69

V. LIMITATIONS

Our approach has a few limitations that influence its performance and scalability. The system relies mostly on Gardiner codes and their pre-defined mappings to English meanings. This makes the translation process very dependent on the completeness and accuracy of the mapping dataset and incorrect mapping of any code may lead to translation errors or incomplete results, especially in cases of missing or inaccurately defined codes.

The second limitation is that the dataset size is small and thus may lead to over-fitting. The model majorly memorized the provided training data but could not generalize into unseen examples. Considering this fact, a core limitation is posed against the model's adaptation capability and robustness.

Another limitation is that the system cannot go further in deep semantic or contextual understanding. Translations are often literal and cannot capture subtle or complex meanings, as is quite common in hieroglyphic texts. The hieroglyphs are not just individual signs, their meaning depends upon their combination and the overall context in which they appear. This leads to a limited translation, which in no way reflects the depth and complexity of meaning in the original text.

Finally, a scalability issue is present as the dataset structure is dictionary-based. The approach was designed for the Gardiner codes present in the dataset and would require significant additional work to extend to be able to generate sentences. This involves creating new mappings for symbols and fine-tuning the model with an expanded dataset that is not available. Because of that, the system does not easily adapt to other hieroglyphic systems or broader linguistic contexts, and its use and scalability for general tasks in hieroglyph translation are therefore limited.

VI. CONCLUSION

This paper presents a comprehensive AI-driven approach for translating Egyptian hieroglyphics into English, combining structured symbolic mapping with state-of-the-art natural language processing techniques. Using Gardiner's code as a linguistic anchor, we explored three methods: fine-tuning a transformer model (T5-small), prompt engineering with LLaMA 3.2, and a retrieval-augmented generation strategy. Our manually curated and rigorously preprocessed dataset enabled reliable training and evaluation across multiple metrics, including Accuracy, Precision, Recall, F1 Score, BLEU, and ROUGE-L.

Among the approaches, prompt engineering emerged as the most efficient and accurate, achieving perfect performance without additional training. Fine-tuning demonstrated strong generalization ability with an F1 Score of 0.7374 and BLEU score of 34.40, while the RAG model offered flexible, context-aware translations with an F1 Score of 0.84. These results underline the potential of large language models to bridge the gap between ancient symbolic writing and modern natural language.

Despite these successes, the system's reliance on Gardiner's code, limited dataset size, and lack of deep semantic representation pose challenges for broader applicability. Future work should focus on expanding the dataset, incorporating contextual co-occurrence of glyphs, and exploring multi-modal translation pipelines that fuse visual features with symbolic and textual embeddings. By combining historical linguistic structure with the capabilities of modern AI, this study contributes to both digital humanities and applied NLP, opening new avenues for preserving and understanding ancient languages.

REFERENCES

[1] A. H. S. Gardiner, *Egyptian Grammar: being an introduction to the study of hieroglyphs*. 1 1927.

[2] A. Barucci, C. Canfailla, C. Cucci, M. Forasassi, M. Franci, G. Guarducci, T. Guidi, M. Loschiavo, M. Picollo, R. Pini, L. Python, S. Valentini, and F. Argenti, *Ancient Egyptian Hieroglyphs Segmentation and Classification with Convolutional Neural Networks*. 1 2022.

[3] S. E. Mohsen, R. Mansour, A. Bassem, B. Dessouky, S. Refaat, and T. M. Ghanim, "Aegyptos: Mobile Application for Hieroglyphs Detection, Translation and Pronunciation," *2023 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, pp. 1–8, 9 2023.

[4] R. Elnabawy, R. Elias, and M. Salem, "Image Based Hieroglyphic Character Recognition," *2018 14th International Conference on Signal-Image Technology Internet-Based Systems (SITIS)*, vol. 7, pp. 32–39, 11 2018.

[5] A. Sobhy, M. Helmy, M. Khalil, S. Elmasry, Y. Boules, and N. Negied, "An AI based automatic translator for Ancient Hieroglyphic Language—From scanned images to English text," *IEEE Access*, vol. 11, pp. 38796–38804, 1 2023.

[6] M. Franken and J. C. Van Gemert, "Automatic Egyptian hieroglyph recognition by retrieving images as texts," *MM '13: Proceedings of the 21st ACM international conference on Multimedia*, pp. 765–768, 10 2013.

[7] R. Moustafa, F. Hesham, S. Hussein, B. Amr, S. Refaat, N. Shorim, and T. M. Ghanim, "Hieroglyphs Language Translator using deep learning techniques (Scriba)," *2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, pp. 125–132, 5 2022.

[8] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 2963–2970, 6 2010.

[9] E. Roman-Rangel, C. P. Gayol, J.-M. Odobez, and D. Gatica-Perez, "Searching the past," *Proceedings of the 30th ACM International Conference on Multimedia*, vol. 25, pp. 163–172, 11 2011.

[10] M. De Cao, N. De Cao, A. Colonna, and A. Lenci, "Deep Learning Meets Egyptology: a Hieroglyphic Transformer for Translating Ancient Egyptian," *Proceedings of the 1st Workshop on Machine Learning for Ancient Languages (ML4AL 2024)*, pp. 71–86, 1 2024.

[11] P. Wiesenbach and S. Riezler, "Multi-Task Modeling of Phonographic Languages: Translating Middle Egyptian Hieroglyphs," *Proceedings of the 16th International Conference on Spoken Language Translation*, 11 2019.

[12] J. Kreutzer, J. Bastings, and S. Riezler, "Joey NMT: A minimalist NMT toolkit for novices," *arXiv (Cornell University)*, 1 2019.

[13] N. C.-L. Liu, S. Jaeger, and M. Nakagawa, "Online recognition of chinese characters: the state-of-the-art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 198–213, 2 2004.