A Generative Model for the Mycenaean Linear B Script and Its Application in Infilling Text from Ancient Tablets

KATERINA PAPAVASSILEIOU and DIMITRIOS I. KOSMOPOULOS, University of Patras
GARETH OWENS, Hellenic Mediterranean University

We present a generative neural language model for the most ancient proven stage of the Greek language, the Mycenaean Greek attributed by the Linear B script. To capture the statistical structure of the Mycenaean documents, we present a Bidirectional Recurrent Neural Network and compare it to the traditionally used *n*-grams. The model is used to supplement the damaged parts of the Mycenaean texts, namely the incomplete, to a greater or lesser extent, words, which are typically discovered on partially damaged clay tablets. We verify our method experimentally using ground-truth, then we demonstrate our results on real cases and compare with experts' opinions. We also present a methodology to augment our dataset, which turns out to improve our results.

CCS Concepts: • **Computing methodologies** → **Supervised learning**; **Language resources**; **Neural networks;**

Additional Key Words and Phrases: Dataset, Linear B script, recurrent neural networks

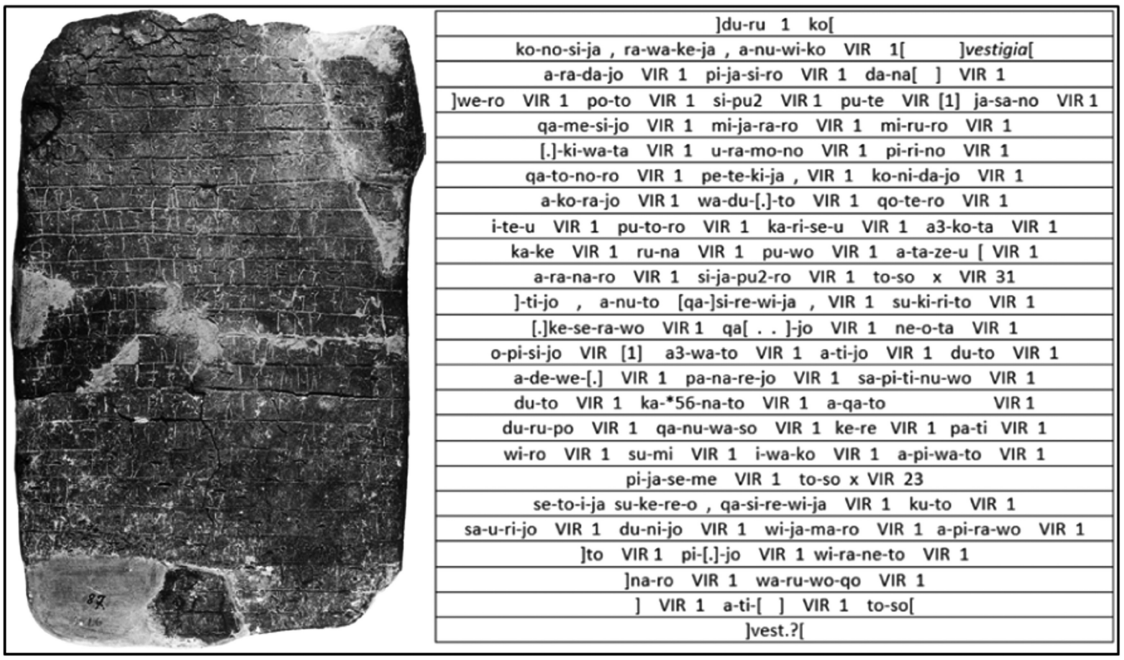
ACM Reference format:
Katerina Papavassileiou, Dimitrios I. Kosmopoulos, and Gareth Owens. 2023. A Generative Model for the Mycenaean Linear B Script and Its Application in Infilling Text from Ancient Tablets. *ACM J. Comput. Cult. Herit.* 16, 3, Article 52 (August 2023), 25 pages.
https://doi.org/10.1145/3593431

1 INTRODUCTION

The Mycenaean Linear B script constitutes one of the writing systems used in the Aegean region in the second half of the second millennium B.C.E. It is a syllabic script deciphered by Michael Ventris [1, 2] and announced on 1-6-1952. He proved that the syllables of the Linear B script form words of the Greek language and that the Mycenaean world was both linguistically and culturally linked to ancient Greece. The Mycenaean Linear B texts, though brief, provide valuable information about the development and evolution of the Greek language throughout the centuries. However, the importance of Mycenaean documents is of interest, not only for linguistics and epigraphy, but also for our understanding of ancient religion and economies, sociolinguistics, literacy studies, and ancient history, since those texts may provide information on the political and administrative organization, the social structure, the economic activity, the religion, and the military organization of the Mycenaean kingdom [3–5].

This work is partially supported by the Greek Secretariat for Research and Innovation and the EU, Project SignGuide: Automated Museum Guidance using Sign Language T2EDK-00982 within the framework of "Competitiveness, Entrepreneurship and Innovation" (EPAnEK) Operational Programme 2014–2020.
Authors' addresses: K. Papavassileiou and D. I. Kosmopoulos (corresponding author), University of Patras, Patras GR-26500, Greece; emails: {cpapavas, dkosmo}@upatras.gr; G. Owens, Hellenic Mediterranean University, Herakleion GR-71410, Greece; email: ogareth@hmu.gr.

1556-4673/2023/08-ART52 $15.00
https://doi.org/10.1145/3593431



]du-ru 1 ko[
ko-no-si-ja , ra-wa-ke-ja , a-nu-wi-ko VIR 1[ ]*vestigia*[
a-ra-da-jo VIR 1 pi-ja-si-ro VIR 1 da-na[ ] VIR 1
]we-ro VIR 1 po-to VIR 1 si-pu2 VIR 1 pu-te VIR [1] ja-sa-no VIR 1
qa-me-si-jo VIR 1 mi-ja-ra-ro VIR 1 mi-ru-ro VIR 1
[.]-ki-wa-ta VIR 1 u-ra-mo-no VIR 1 pi-ri-no VIR 1
qa-to-no-ro VIR 1 pe-te-ki-ja , VIR 1 ko-ni-da-jo VIR 1
a-ko-ra-jo VIR 1 wa-du-[.]-to VIR 1 qo-te-ro VIR 1
i-te-u VIR 1 pu-to-ro VIR 1 ka-ri-se-u VIR 1 a3-ko-ta VIR 1
ka-ke VIR 1 ru-na VIR 1 pu-wo VIR 1 a-ta-ze-u [VIR 1
a-ra-na-ro VIR 1 si-ja-pu2-ro VIR 1 to-so x VIR 31
]-ti-jo , a-nu-to [qa-]si-re-wi-ja , VIR 1 su-ki-ri-to VIR 1
[.]ke-se-ra-wo VIR 1 qa[. .]-jo VIR 1 ne-o-ta VIR 1
o-pi-si-jo VIR [1] a3-wa-to VIR 1 a-ti-jo VIR 1 du-to VIR 1
a-de-we-[.] VIR 1 pa-na-re-jo VIR 1 sa-pi-ti-nu-wo VIR 1
du-to VIR 1 ka-*56-na-to VIR 1 a-qa-to VIR 1
du-ru-po VIR 1 qa-nu-wa-so VIR 1 ke-re VIR 1 pa-ti VIR 1
wi-ro VIR 1 su-mi VIR 1 i-wa-ko VIR 1 a-pi-wa-to VIR 1
pi-ja-se-me VIR 1 to-so x VIR 23
se-to-i-ja su-ke-re-o , qa-si-re-wi-ja VIR 1 ku-to VIR 1
sa-u-ri-jo VIR 1 du-ni-jo VIR 1 wi-ja-ma-ro VIR 1 a-pi-ra-wo VIR 1
]to VIR 1 pi-[.]-jo VIR 1 wi-ra-ne-to VIR 1
]na-ro VIR 1 wa-ru-wo-qo VIR 1
] VIR 1 a-ti-[ ] VIR 1 to-so[
]vest.?[

Fig. 1. On the left, the image of the Mycenaean tablet KN As 1516 (image copyright belongs to the Hellenic Ministry of Culture and Sports - "Hellenic Organization of Cultural Resources Development"). On the right, the transliterated text of the KN As 1516. The illegible parts are indicated by square brackets.

The Mycenaean Linear B script is recorded mostly on clay tablets [6, 7] (c. 5500 have been recovered to date). The main difficulties in studying these clay inscriptions are: (a) Their eminently administrative content. Many of the Mycenaean tablets are made up of simple lists, revealing less than expected about their syntactic structure. (b) Their subject, which complicated the decipherment of the Linear B script. As they deal extensively with people and places, inevitably a large number of words contained in the tablets are proper nouns. While the human names were the starting point for deciphering the Persian cuneiform script, as they concerned the already known names of Persian kings, they could not offer help in the case of Linear B, where they concerned the names of completely humble people [8]. (c) Their state of preservation, since most of them are broken, worn, or burned, to a lesser or greater extent. Due to physical damage, many Mycenaean syllables from these documents are missing (see Figure 1). Therefore, notable pieces of our ancient cultural heritage may have been lost. Although the Mycenaean texts are mostly lists of supplies entering, leaving, or stored in palaces and telegraphic inscriptions of goods, they are useful as primary sources for the economy, trade, religion, social stratification, and administrative organization of the Mycenaean Greece.

Currently, the efforts to study and restore the Mycenaean tablets have been done manually by a handful of experts [2, 6, 9–19]. It is a long and time-consuming process, which requires extensive knowledge in the specific genre and corpus. When it comes to restoration, for the resulting infillings it is hard to quantify the degree of certainty for each one of them. To mitigate those issues, it would be ideal to have an automated system to assist the experts by recommending possible infilling alternatives of the missing parts and by sorting the probabilities of each of those recommendations, based on the available corpus. In this article, we propose such a tool for infilling Mycenaean Linear B damaged tablets.

This article contributes by presenting a generative model for the Mycenaean Greek language attested by ancient tablets. To this end, we train a **language model** (**LM**) using a Recurrent Neural Network (and namely a

symbol-level RNN, where a symbol corresponds to a Mycenaean syllable or a Mycenaean logogram). We use data that we have collected and curated using the available textbooks [20] and the electronic corpora [21]. Furthermore, we exploit this model to predict the missing syllables on tablets that are partially damaged.

The rest of this article is organized as follows: In Section 2, we present the related work regarding the efforts made on other languages and the related methods. In Section 3, we present the dataset that we collected for our experiments. In Section 4, the data augmentation techniques are described in order to make our dataset sample larger. Section 5 demonstrates the proposed methodology for language modeling and for missing symbol recovery. Section 6 shows the experimental results. Finally, in Section 7, we present our conclusions and the future work.

2 RELATED WORK

Our work is related to the concept of language modeling. A generative LM calculates the probability of occurrence of a particular sentence/word/character within a specific context/topic, by analyzing a body of similar training texts in the same context or topic. In this way, statistical and probabilistic techniques are developed capable of predicting and determining the probability of occurrence in a sequence, if, for example, the LM operates at the level of words (sentences, characters, etc.). To this end, **recurrent neural networks** (**RNNs**) have been used in a generative fashion at the word level, e.g., [22], at character level, e.g., [23], or hybrid, e.g., [24]. Such models though valuable for text generation, do not typically consider the text after the missing part, which may contain valuable information.

Some of the simplest models are the n-gram models. These are probabilistic models for predicting the next item in a sequence of n elements and can be used to model almost any type of sequential data. They have been used for machine translation [25], but also for ancient documents restoration [26]. The benefits of n-gram models are their simplicity and scalability. With larger n, a model can store more context, enabling small experiments to scale up. However, when n increases, the out-of-vocabulary n-grams increase noticeably and actually undermine the performance of the model. Therefore, the n-grams are not appropriate to model long dependencies, and more elaborate models are needed.

The problem of text infilling (i.e., the task of predicting missing spans of text - typically whole words) has been reported in the literature for contemporary texts, for which typically large volumes of data are available. It is the generalization of the cloze task, in which the goal is to find the missing word by using context [27]. Frameworks like BERT [28] that are based on the Transformer network can consider previous and following text, but require large volumes of data and fixed span. Fedus et al. use GANs requiring a fixed span as well [29]. Donahue et al. present an infilling framework that is able to capitalize on existing LMs [30]. It has been tested on different domains such as short stories, scientific abstracts, and lyrics. Similar methods that update their estimate in an iterative way are presented in [31] and [32]. Shen et al. propose a **Blank Language Model** (**BLM**) which looks ideal for text generation tasks [33]. BLM generates the missing fragments by jointly modeling context and missing content. It supports the control of generation location and produces consistent infilling of variable length. BLM is testing on three experiments: text infilling [31], ancient text restoration [34], and style transfer [35]. In all experiments, the sequence representations in BLM are obtained using the encoder module of transformer base [36]. In text infilling task, they experiment on a dataset which has 100K/10K/10K documents for train/valid/test, respectively, with a maximum length of 200 words for each document. A lot of data is required for the success of the model as well.

The problem of text restoration through infilling in ancient texts is attracting more and more attention from the research community. Though the results from the **Natural Language Processing** (**NLP**) community have been only partially applied, obviously due to the lack of sufficient data. Some of the most representative such works are briefly described in the following.

Roued-cunliffe uses a decision support system called DUGA for reading ancient documents in the Latin language found in Vindolanda (Britain) [37]. She uses the so-called cruciverbalistic approach: It begins by

establishing the letters that are legible and uses them as a foundation for a subsequent hypotheses. A knowledge-base of previously interpreted documents from the same period is used to extract word lists and frequencies. These are then used to suggest different interpretations of words and letters, as well as missing parts, using a hierarchical approach from individual symbols to whole sentences. The system is therefore largely based on the experts' decisions. Kang et al. present a multi-task learning approach based on the Transformer networks to effectively restore and translate ancient historical documents based on a self-attention mechanism, specifically utilizing two Korean historical records, one of the most voluminous historical records in the world [38]. This work combines three different studies: the restoration of damaged documents (recovering), neural machine translation (translating), and the analysis of historical records (mining). The proposed model consists of embedding and output layers for Hanja and Korean, and three Transformer modules: the shared encoder (for both the restoration and translation tasks), the restoration encoder (for the restoration task), and the translation decoder (for translating Hanja sentences into modern Korean sentences). In order for these tasks to succeed, it is necessary using a large-scale training corpus.

Similar to our work is the PYTHIA system [34] and its follow-up system Ithaca [39]. It aims to fill the missing symbols (characters) in ancient Greek inscriptions. The authors use a sequence-to-sequence framework [40] with **Long Short-Term Memory** (**LSTM**) networks in the encoder and the decoder. The encoder involves the input character embeddings sequence with missing characters, and a separate stream is also modeled using the word sequence as embeddings as well; an attention layer is also used. The decoder is trained to output the missing characters. They use a dataset that results from processing the epigraphical corpora of the Packard Humanities Institute [41], the PHI-ML. As Shen et al. argue in their ancient text restoration experiment [33], Assael et al. perform restoration at the character-level where the number of characters to recover is assumed to be known and indicated by a corresponding number of "?" symbols [34]. In reality, when epigraphists restore a deteriorated document, the length of the lost fragment is unknown and needs to be guessed as a first step. BLM, in essence a variant of BLM, the L-BLM, can bypass this limitation and flexibly generate completions without this additional knowledge. A single token, sized equal to the number of "?" symbols, is defined and the L-BLM is trained to predict a character to fill in and the length of the new blank to its left. Compared to our work, the problem presented by the authors of these articles [33, 34] is similar in the sense that it concerns a known script and known language and uses a machine learning architecture. However, our task is more challenging, due to the fact that our corpus is of much smaller size (over 40,000 inscriptions available in the aforementioned articles versus 1,100 inscriptions in ours). This means that sequence-to-sequence architectures, requiring the calculation of many weights, cannot be effectively trained in our case.

Fetaya et al. use recurrent neural networks (LSTMs) to restore fragmentary Babylonian texts [26]. These involve ancient texts in the Akkadian language, which belong to the Semitic language family. Comparisons to simple 2-gram baseline approach (considering the previous and the next word) are made, resulting in better performance. The experiments use a dataset of 3,000 transliterated archival documents belonging to economic, juridical, and administrative genres. Similarly to this work, Lazar et al. also introduce BERT-based models aiming to solve the task of predicting missing signs in Akkadian texts [42]. The difference with the previous article [26] is that the completion of missing signs is done by combining large-scale multilingual pretraining with Akkadian language finetuning. Although Fetaya et al. have little data at their disposal to train the learning algorithm (c.3000 Babylonian transliterated texts, 539–331 B.C.E.), much more than ours (c. 1100 Mycenaean clay tablets, 1400–1200 B.C.E.), but unquestionably a small amount of data, what is emphasized by the authors is that the late Babylonian texts are a genre with highly structured syntax. These are structured official bureaucratic documents, e.g., legal proceedings, receipts, promissory notes, contracts, and so on. This is in stark contrast to our own work, which is a task significantly impeded by syntactic inconsistencies. This is proved by the fact that the pre-processing of our data, i.e., the management of each Mycenaean tablet [43], is not a simple matter but require special handling, so as to extract valid sequences of Mycenaean words in accordance with the principles of Mycenaean language.

There are also several works dealing with translation or decipherment of ancient languages, which are not directly related to our work, but could potentially benefit from it, since they assume no missing symbols. Some indicative such works are given in the following. Snyder et al. demonstrate automatic decipherment by using a Bayesian network that incorporate linguistic constraints using non-parallel corpora of Ugaritic and Hebrew [44]. Then, Berg-Kirkpatrick and Klein use a combinatorial optimization method on the same data [45]; they present a simple objective function to solve the decipherment and cognate pair identification problems. Luo et al. try to do decipherment of Ugaritic and Linear B using non-parallel corpora, by aligning words from the ancient language to words of the known one [46]; they employ a sequence-to-sequence model (biredictional-LSTM as the encoder and a single layer LSTM as the decoder) with additional linguistic constraints. However, Luo et al. in their article [47] argue that the methods described in [46] are based on assumptions that are not applicable to many undeciphered scripts and they propose a decipherment model to extract cognates from undersegmented texts, without assuming proximity between lost and known languages. The model is evaluated on both deciphered languages (Gothic, Ugaritic) and an undeciphered one (Iberian). Page-Perron et al. aim for the translation of ancient Sumerian texts to English using neural machine translation and a parallel corpus of about 3,000 texts [48]. Another attempt to implement automatic translation into cuneiform languages is presented in the article [49]. Punia et al. are experimenting with various architectures and claim that using pretrained word embeddings in sequence-to-sequence models with attention can achieve the best performance in the sparse data setting. Finally, Park et al. propose the first ancient Korean neural machine translation model using a Transformer [50]. They also contribute by suggesting the Share Vocabulary and Entity Restriction BPE (SVER BPE), which is a new subword tokenization method based on the characteristics of ancient Korean sentences. They claim that SVER BPE achieves better results than conventional subword tokenization methods (like BPE and SentencePiece).

In our previous article ([43]), we describe the methodology to create a preliminary version of our Mycenaean dataset comprising text sequences of the Mycenaean Greek language. We have set up a preliminary experiment using a discriminative model, namely a Conditional Random Field to predict the missing parts. In that work, we do not consider the information after the gap and we do not quantify the certainty level, as well as the other valid options. Here, we are addressing these drawbacks. Furthermore, we provide a data augmentation method to handle the lack of data.

3 THE DATASET

We will provide a brief description on how the sequences from the Mycenaean inscriptions were extracted and why specific standards were introduced in order to create the Mycenaean dataset. For more details, the interested reader can refer to our previous work [43].

Considering the absence of strict syntactic rules in the Mycenaean language [8, 17], it might seem difficult to extract any structure out of the Mycenaean inscriptions/documents. However, the experts have argued that there is still some structure that can be exploited, which is closely related to the way that we have to read them [8, 51]. The way of creating the sequences, and how the Mycenaean tablets are read, is a key factor in extracting the Mycenaean language structure and in capturing the meaning. The modeling task may be crucially impeded though, by the syntactic inconsistencies of the language, as well as by distinctions among scribes.

Here, we present briefly the Linear B script and its transcription. The Mycenaean Linear B script uses two basic symbol systems, one phonetic and one logographic. The symbols of the phonetic system are called syllabograms-syllables. The phonetic system is usually represented transcribed, i.e., the syllable is rendered in letters, and in most cases by a combination of consonant and vowel. The syllabary, namely the system of the phonetic symbols, includes at least 87 different syllables. Of these, a very limited number (about 14) has not yet been transcribed, i.e., its phonetic value is not known with certainty. José L. Melena writes in [52] that their resistance to decipherment is due to the scanty evidence available and/or to the fact that they appear mainly in personal names and place-names of non-Greek origin. For the symbols of the logographic system, the term ideograms is usually used. Such

symbols need not be representations of the objects themselves and clearly do not refer directly to ideas, events, or concepts but rather to words of the Mycenaean language. Hence, the term logogram is preferable to the widely used, and less accurate term, ideogram [52]. The logographic signs are 143 in number. For the representation of logograms a conventional transcription is used, based mainly on the abbreviation of the Latin name of the represented object or being, e.g., VIR "man", MUL(ier) "woman". If the ideogram has not been identified, then the abbreviation is written with an asterisk followed by a three-digit number. In addition to the phonetic symbols and the ideograms, there are also numbers that follow the decimal system and measurements units of weight and capacity [51, 53].

The Mycenaean dataset contains sequences of Mycenaean words, of typical length of 6–10 syllables, in transcription, i.e., sequences of phonetic values as they have been identified and attributed to the syllables of the Mycenaean Linear B script. In order to generate this dataset, we used the available Linear B texts in the related literature: [4, 6, 8, 17, 20, 51, 54, 55]. These works refer to texts mostly written on clay tablets (but also on clay vases, stamps, and tags), and were found throughout Mycenaean Greece (Knossos, Khania, Pylos, Mycenae, Tiryns, Sparta, and Thebes).

The separation of the Mycenaean sequences was based on the following observations. The divider (short upright line on the tablets, conventionally represented by a comma on the transliterated texts) is a word-separation symbol. Furthermore, we assumed that numeric and metric signs are followed by new sequences. This way, we sorted the signs which occur in groups, i.e., the words, as well as the ideograms. The numeric signs and measurement units of weight and capacity were excluded from the dataset. An example of sequences created by two Mycenaean tablets is displayed in Figure 2.

Records of artifacts, livestock, staples, and so on are only rarely indicated by words syllabically written; mostly the scribes use symbols, which stand for the commodity in question, the logograms (or ideograms) [51, 53]. In some cases, the ideograms are presented after the word they describe, confirming and identifying the information provided by the tablet. In many other cases, the information is derived exclusively from the ideograms. So, the semantic value of the Mycenaean ideograms plays an important role, as it did in the decipherment of the Linear B script, and could not be missing from our database of sequences of Mycenaean words.

The Linear B inscriptions are classified into categories, the so-called series, depending on the object to which they refer; many of these categories are subdivided in order to state classifications of different content [8]. Usually the tablets that belong to the same series have a similar structure. This is also reflected on the sequences of our dataset that come from the same or different series. For example, Figure 2 shows the sentences resulting from two tablets belonging to different series, namely series D and series L. They appear to be similar, i.e., both involve a group of large-sized syllables (words) on the left side of the tablet and two rows of groups of smaller syllables (words), usually separated by a dividing line, on the right side; however their content is rendered differently. According to [51], the text from the tablets of series D of Knossos, starts from the second line and continues on the first, as shown in Figure 2 (left). On the other hand, from the tablet of series L emerge two sequences, since the word in large syllables refers to both rows of the right part of the tablet.

The reason we chose to build a symbol-level LM instead of a word-level one, that could also be valid, is related to the nature of the text corpus we have at our disposal. It is important to note that the Mycenaean clay tablets were annual draft censuses typically used for daily economic-administrative notes, and were destroyed after a period of time (financial year) to reuse clay for new recordings. As a consequence, the surviving documents in Mycenaean linear B are a minimal part of the total of written testimonies, since they are believed to date back mainly from the last year before the destruction of the Mycenaean administrative centers. So, in this last annual census of Mycenaean palaces, it was rather hard to find repeating words, due to the large vocabulary and due to the limited corpus size. On the other hand, syllables are far fewer and frequently repeated and thus easier to model. The predominantly economic necessity that prompted the invention of writing on Crete makes the inscription rather brief and terse, often resulting in sequences consisting of even single words. Such one-word sequences would not be very helpful in a word-level LM. In summary, to infill the damaged parts of the

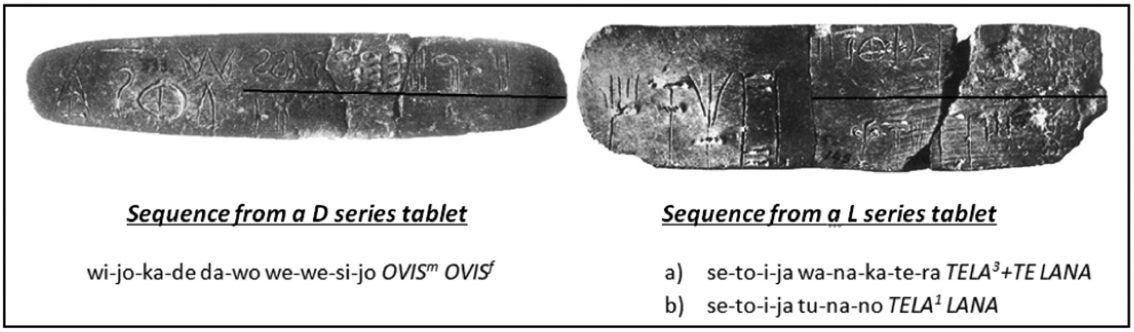


Fig. 2. Mycenaean sequences from the series D (left the image of the Mycenaean tablet KN Db 1155 + 5378 + 5688) and L (right the image of the Mycenaean tablet KN Lc 525) (the copyright of the images belongs to the Hellenic Ministry of Culture and Sports - "Hellenic Organization of Cultural Resources Development"). Translation of the left Mycenaean tablet (the phonetic version of the words is provided in parentheses, as they would be pronounced in the then Mycenaean Greek) *"Wi-jo-ka-de (Wiokados - human name, the shepherd): at Da-wo (Dawos - place name) collection of We-we-si-jo (Werwesios - human name, the collector), eighty-six rams and fourteen ewes"*. Translation of the right Mycenaean tablet *"From Se-to-i-ja (Setoijai - place name): 40 carpets of royal type (wanaktera), 100 measures of wool and 3 cloths of tu-na-no type (the meaning of the word "tu-na-no" is unknown, may not be Greek).*

Mycenaean tablets we preferred our model to operate at the level of symbols, namely the Mycenaean syllables, which form words; these syllables, in turn, form the Mycenaean sequences.

To build a symbol-level LM and use it to generate Mycenaean texts, it is first necessary to define a training set comprising a corpus of Mycenaean sequences. In our case the training set, in the experimental parts, consists of the tablets of series D. The next step is the tokenization, i.e., the process during which the sequence is split into its constituent elements, in our case symbols corresponding to syllables and logograms (ideograms), and for each token a unique identifier is assigned. Each token is represented by a one-hot encoding vector including the space token. Consider the sequence "wi-jo-ka-de da-wo we-we-si-jo $OVIS^m$ $OVIS^f$" resulting from the series-D Mycenaean tablet in Figure 2. The tokenization of this Mycenaean sequence, namely its map to the individual symbols in the Vocabulary, leads to 16 input tokens (see Table 1).

The corpus consists of the tablets of "series D" of Knossos, which is the largest classification; it comprises the accounts of the sheep herds of Crete, in around 1,100 tablets. The fact that the origin of these tablets (i.e., the center in which they were found), in contrast to the documents of the other series, is unique (Knossos), as well as the fact that the majority of them (c. 686–720 tablets) are probably written by the same scribe (Hand 117 [21]), serve in their best possible processing but mainly in the application of the augmentation techniques. For this reason, as a first step, we choose a single category of tablets and prefer not to mix many categories together to exploit the homogeneity observed in the structure of a single category. The repetition of similar patterns is important for machine learning.

From those tablets we extracted 513 complete sequences, without missing syllables, which was the training set of the model. The remaining sequences were more or less damaged, making about 145 Mycenaean sentences, which were hard to read. We defined a Vocabulary of [77 syllables, space, 23 ideograms]. The most frequent ideograms in this series were: $OVIS^f$ = female sheep, $OVIS^m$ = ram, and LANA = wool.

Most of the time the syllables are placed one after the other so as to form recognizable words. But they can also be used in other ways, as for example individually. When a syllable is used individually, it operates either as a *ligature* or an *adjunct* or as an abbreviation of a word. It is common for an ideogram to join with a syllable (*ligature*). Ligatures do not appear in the D series we are dealing with. A syllable can be placed on top of an ideogram, thus forming a syllabic definition (*adjunct*). A common such example in the D series is the $neOVIS^m$

Table 1. Example Sequence and Its Tokenization Including Syllables and Ideograms

wi-jo-ka-de da-wo we-we-si-jo $OVIS^m$ $OVIS^f$															
wi	jo	ka	de		da	wo		we	we	si	jo		$OVIS^m$		$OVIS^f$

which identifies the animal as young (*ne-wo*). Finally, in the category of our interest there are several words that are abbreviated and refer to the animals and wool, i.e., exclusively to the ideograms of the category. For example, $paOVIS^m$, $peOVIS^m$, and *oLANA*, where the syllables *pa*, *pe,* and *o* are abbreviations of the words *pa-ra-jo/ja* (='old'), *pe-ru-si-nu-wo* ('last year's'), and *o-pe-ro* ('dept') correspondingly. All these cases contribute to the creation of 19 additional ideograms beyond the three basic.

4 DATA AUGMENTATION

It is generally known that the amount of data used for training impacts the model performance. In our case, the sequences resulting from Mycenaean texts are rather scarce especially when we limit our model to a single series, such as series D (see Section 6). Augmentation methods, namely methods used to generate additional, synthetic data using existing data, are also gaining popularity in NLP applications [56]. Here, the data augmentation has to be done carefully due to the structure and the semantics of the language.

In order to augment our training dataset and make the model generalize better, we provide the following methods-rules, fully interwoven with the structure of the Mycenaean tablets and the Mycenaean reality [2, 6, 7, 51], i.e., the place names (toponyms), the professions, the institutions, and the hierarchy of the Mycenaean society.

The rules analyzed below concern series D and were used to conduct the experiments of Section 6:

(1) Most of the tablets of series D show the toponym (*"PN"*) of the administrative area together with the name of the responsible shepherd (*"MN"*). Sometimes a second name appears that is of the wool collector, more specifically of the owner of the herd (*"mn"*) [6, 51]. So, most tablets follow the structure:

MN (shepherd)	*PN* (toponym)	*mn* (owner/collector)

In some cases, a slightly different structure appears:

MN (shepherd)	*mn* (owner/collector)	*PN* (toponym)

So, the first rule concerns the swapping of words referring to the toponym and the owner. This can only happen if and only if the Mycenaean tablets embody these words, which represent these names (word categories). The method of swapping the specific words produced 94 more sequences.

(2) A possible link has been established between some toponyms by their simultaneous appearance as a pair on some cumulative tablets (the Dn collection of series D is a sum collection) [8, 51]. These are given in the following:

Locations close to each other								
pa-i-to	e-ko-so	da-*22-to	ti-ri-to	ra-ja	do-ti-ja	ri-jo-no	ru-ki-to	u-ta-no
da-wo	su-ki-ri-ta	*56-ko-we-i	qa-ra	pu-na-so	ra-su-to	ra-to	pu-so	qa-mo

When these adjacent areas appear from the Mycenaean texts to belong to the same owner, then we consider safe proceed to mutual substitutions. From this toponyms' substitutions, we produced 21 additional sequences.

(3) Sometimes the standard structure of series D tablets changes either because the owner's name does not appear (see tablets in Figures 7 and 8), or because the place name is omitted. So, another technique to augment our dataset is to select these words from a Mycenaean sequence that follows the standard structure

and remove one at a time so as to create two new sequences for each case. With the word deletion rule, 170 new sequences were produced.

(4) The fourth rule concerns the abridged structure

MN (shepherd)	*PN* (toponym)

On these Mycenaean tablets, the owner's name is omitted. In any such case, we can create as many sequences as the owners of each toponym, resulting from the tablets that follow the normal structure. This method of owner insertion resulted in 440 new sequences.

These augmentation rules offer 725 additional samples. The maximum number of rules applied to a sequence is four. This means that they are generated up to four new sequences from each original one. On the other hand, there are sequences that are not addressed by any augmentation rule. Therefore this process introduces bias in the estimation, since through augmentation some sequences practically take more weight than others. To alleviate that, we introduce some random repetitions of the sequences or their derivatives through augmentation, so that each sequence or its derivatives appear exactly five times in the dataset. Thus, the duplicate samples presented on the network are 1,327.

5 METHODOLOGY

In this section, we describe the background on the related work we use in our methodology, i.e., RNNs and **Bidirectional Recurrent Neural Networks** (**BRNNs**), to train generative models of the Mycenaean sequences.

The RNNs are artificial neural networks where connections between nodes form a directed graph along a temporal sequence. Therefore, the graph can represent temporal dynamic behavior [57]. The RNNs can use their internal state to process sequences of variable length, and therefore they are popular for doing tasks such as speech recognition, video processing, machine translation, and so on.

The RNNs are plausible for learning sequential data models like text documents. However, they suffer from short-term memory which can not capture the long-range dependencies that may exist in large sequences. Their gradient shrinks noticeably as it back-propagates through time (vanishing gradient); when it becomes too small, the gradient does not really contribute to model learning. In our case, however, the Mycenaean documents contain rather short sequences that do not typically exceed four words; therefore, the RNNs are not expected to suffer seriously from such issues in our application. The more advanced versions of sequential models like the LSTMs [58] and the **Gated Recurrent Unit** (**GRUs**) [59] include internal mechanisms (gates) that overcome the aforementioned problem and allow them to model effectively much longer sequences; however, in short sequences these mechanisms introduce rather unnecessary overhead, i.e., higher number of parameters that need to be learnt. To learn so many parameters without overfitting requires larger amount of representative training data, which are very difficult to obtain in applications like ours.

Given the typical length of the Mycenaean sequences and the size of our dataset, we built our models based on a simple RNN architecture. That architecture is presented in Figure 3, where the following notations are used:

— $\mathbf{x}_t \in \{0,1\}^d$ is the input vector at timestep t, where d is the vocabulary size.
— $\boldsymbol{\alpha}_t \in \mathbb{R}^m$ is the activation vector that stores the values of the hidden units at timestep t, m is the number of hidden units.
— $\mathbf{y}_t \in \mathbb{R}^d$ is the output vector of the network at timestep t.
— $\mathbf{W}_{ax} \in \mathbb{R}^{mxd}$ are weights associated with inputs in hidden layer.
— $\mathbf{W}_{aa} \in \mathbb{R}^{mxm}$ are weights associated with hidden units in hidden layer.
— $\mathbf{W}_{ya} \in \mathbb{R}^{dxm}$ are weights associated with outputs from hidden layer/ with hidden layer to output units.
— $\mathbf{b}_a \in \mathbb{R}^m$ is the bias associated with the hidden layer.
— $\mathbf{b}_y \in \mathbb{R}^d$ is the bias associated with the output layer/ is the bias relating the hidden units to the output.

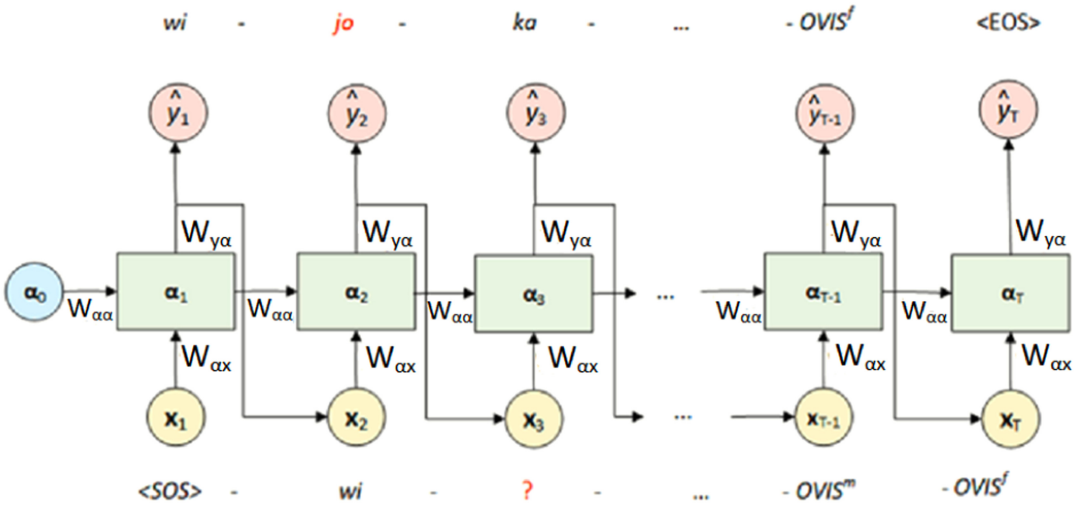


Fig. 3. RNN architecture. The input sequence $[\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_n]$ is the one presented in Table 1 and the missing syllable is "jo". It is predicted as the output of the "next" state.

The RNN at time $t = 1$ computes the activation $\boldsymbol{\alpha}_1$ from input $\boldsymbol{\alpha}_0$, typically set to zero vector, and from $\mathbf{x}_1$, which is set to a predetermined vector signifying the start-of-sequence symbol (<SOS>), as follows:

$$\boldsymbol{\alpha}_0 = \vec{0}, \mathbf{x}_1 = < SOS > \tag{1}$$

$$\boldsymbol{\alpha}_1 = f_1(\mathbf{W}_{\alpha\alpha}\boldsymbol{\alpha}_0 + \mathbf{W}_{\alpha x}\mathbf{x}_1 + \mathbf{b}_\alpha), \tag{2}$$

where f_1 is the $tanh$ function.

Then, we obtain an estimation of the output $\hat{\mathbf{y}}_1$, by sampling a softmax function f_2, which is the pdf of the first symbol as follows:

$$\hat{\mathbf{y}}_1 \sim f_2(\mathbf{W}_{y\alpha}\boldsymbol{\alpha}_1 + \mathbf{b}_y). \tag{3}$$

In the example of the sequence in Table 1 ideally the model should return $\hat{\mathbf{y}}_1 = wi$.

Then, the RNN generates the second symbol given the first one, i.e., $\hat{\mathbf{y}}_2 \sim p(\mathbf{y}_2 \mid \mathbf{x}_1 = < SOS >, \mathbf{x}_2 = wi)$, where the pdf p represents the f_2. In a similar fashion, the RNN generates the third symbol in the sequence given the former two ones, i.e., by sampling $\hat{\mathbf{y}}_3 \sim p(\mathbf{y}_3 \mid \mathbf{x}_1 = < SOS >, \mathbf{x}_2 = wi, \mathbf{x}_3 = jo)$. This logic continues to the last timestep. In this way the RNN generates one symbol at a time going from left to right.

So, for the RNN, it generally applies:

$$\boldsymbol{\alpha}_t = f_1(\mathbf{W}_{\alpha\alpha}\boldsymbol{\alpha}_{t-1} + \mathbf{W}_{\alpha x}\mathbf{x}_t + \mathbf{b}_\alpha), \tag{4}$$

$$\hat{\mathbf{y}}_t \sim f_2(\mathbf{W}_{y\alpha}\boldsymbol{\alpha}_t + \mathbf{b}_y). \tag{5}$$

In order to train this Neural Network, the cost function associated with a single prediction is the following:

$$L_t(\hat{\mathbf{y}}_t, \mathbf{y}_t^0) = -\sum_i y_t^{0,i} log \hat{y}_t^i \tag{6}$$

where $\hat{\mathbf{y}}_t$ is the Neural Network's softmax prediction at time t and $\mathbf{y}_t^0$ is the ground truth one-hot vector. The overall loss for an entire sequence is the sum over all timesteps of the losses associated with the individual predictions:

$$L = \sum_t L_t(\hat{\mathbf{y}}_t, \mathbf{y}_t^0). \tag{7}$$

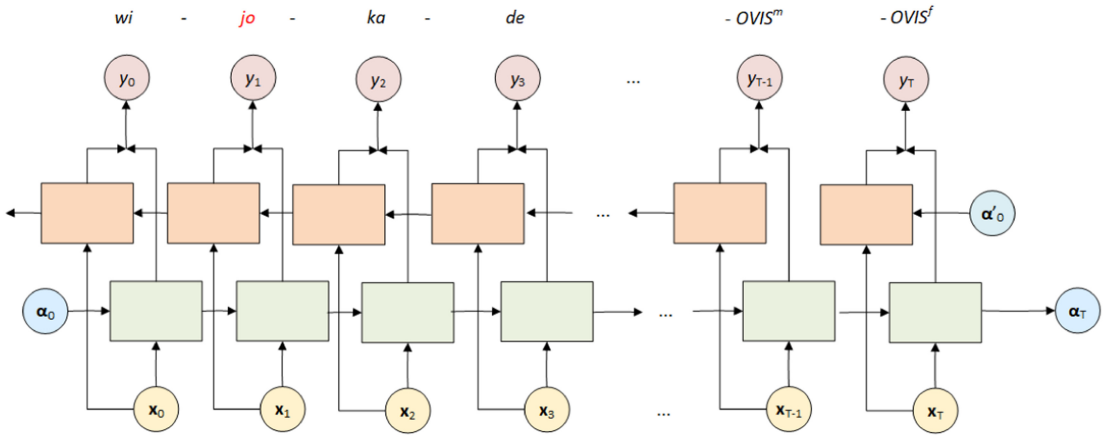


Fig. 4. The BRNN architecture. The input sequence $[\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_n]$ is the one presented in Table 1 and the missing syllable is "jo". It is predicted as the output of $\mathbf{y}_1$ when in the forward direction the input is $\mathbf{x}_1 =$ '*wi*' and in the backward direction the input is $\mathbf{x}_1 =$ '*ka*'.

So, by training this RNN on an appropriate dataset results into a model that it is able to estimate the probability of the next symbol given the past ones. Given a new sequence, it is possible to figure out the probability of the entire sequence by multiplying the probabilities returned by f_2 at each timestep.

A weakness of the unidirectional RNN described above is that the prediction at step t uses information from the inputs earlier $(t-n)$ in the sequence but not information from the inputs later in the sequence $(t+n)$, where $n \in \mathbb{N}$. To figure out what is hidden in the illegible points of a tablet, it is not enough, and sometimes impossible, to just look at the left part of the sequence. We definitely need more information.

The BRNN can model sequences in forward and backward fashion and overcomes the unidirectional RNN limitations. The Figure 4 displays such a network architecture, which defines an acyclic graph; the information flows independently in both directions. The computations from left to right are completely independent of the computations from right to left. So, the following formulas apply:

$$\overrightarrow{\boldsymbol{\alpha}_t} = f_1(\mathbf{W}^f_{\alpha\alpha}\boldsymbol{\alpha}_{t-1} + \mathbf{W}^f_{\alpha x}\mathbf{x}_t + \mathbf{b}^f_\alpha), \tag{8}$$

$$\overleftarrow{\boldsymbol{\alpha}_t} = f_1(\mathbf{W}^b_{\alpha\alpha}\boldsymbol{\alpha}_{t-1} + \mathbf{W}^b_{\alpha x}\mathbf{x}_t + \mathbf{b}^b_\alpha), \tag{9}$$

where $\mathbf{W}^f_{\alpha\alpha}$, $\mathbf{W}^b_{\alpha\alpha}$, $\mathbf{W}^f_{\alpha x}$, $\mathbf{W}^b_{\alpha x}$, $\mathbf{b}^f_\alpha$, $\mathbf{b}^b_\alpha$ are all the model parameters. To get the predictions $\hat{\mathbf{y}}$ in a BRNN, it is necessary to start propagating information from both directions. When we have computed both of the hidden states for a timestep, we can get the prediction $\hat{\mathbf{y}}$ for that timestep using the formula:

$$\hat{\mathbf{y}}_t \sim f_2(\mathbf{W}_y[\overrightarrow{\boldsymbol{\alpha}_t}, \overleftarrow{\boldsymbol{\alpha}_t}] + \mathbf{b}_y). \tag{10}$$

6 EXPERIMENTAL EVALUATION

The goal of this research is to use the generative model for predicting the missing syllables and thus for infilling the missing parts of the Mycenaean tablets. To evaluate the performance of the proposed model, we used the dataset created as described in Sections 3 and 4 (available here [60]) and we conducted two experiments:

(A) Initially we synthetically generated incomplete sequences from complete ones after introducing some gaps, so that we could quantify our results using ground truth data.

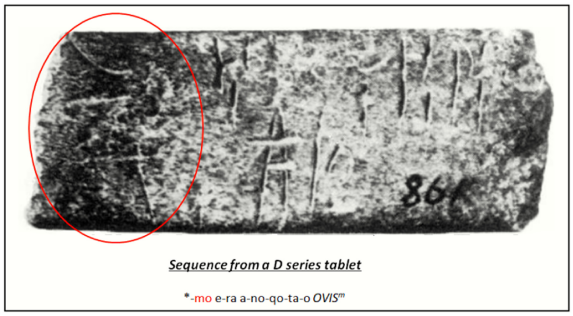


Fig. 5. The image of the Mycenaean Linear B tablet KN Dq 45 (image copyright belongs to the Hellenic Ministry of Culture and Sports - "Hellenic Organization of Cultural Resources Development"). Translation of the Mycenaean tablet *"(missing shepherd's name): at e-ra (place name) belonging to a-no-qo-ta (collector's name), rams"*.

(B) Then we applied our model to predict the missing syllables in some real tablets, which were partially damaged and compared our system's output to the experts' opinion.

In the experiment (A), we did a leave-one-out cross validation. Each test sample/sequence was created by removing a syllable. The special symbol "*" denotes the missing syllable that should be predicted.

In the experiment (B), we used the RNN model trained with all full sequences, augmented data, and duplicates as mentioned in Section 4; we used the selected hyperparameters of (A). Again the special symbol "*" denotes the missing syllable.

It is important to note that in most cases of the damaged edges of the Mycenaean tablets, the number of missing syllables is definitely more than one, without knowing the exact length of the word. As demonstrated in Figure 5, the syllable "mo" is the last of a Mycenaean word located on the cut edge of the tablet. It is difficult to know how many syllables precede the "mo", so in these cases we could predict the probabilities of the previous syllables in a sequential fashion. Since we are dealing with a generative model, there is essentially no problem in examining more gaps in a sequence, whether they are contiguous or scattered in the sequence. The problem, however, lies in the following: (a) in almost all cases we do not know the exact number of missing characters, for example, we do not know how many syllables precede "mo" in Figure 5, unlike the case of the PYTHIA model [34] and (b) it is exponentially more difficult to evaluate the infillings; however someone could use beam search heuristics to focus on the most promising solutions.

6.1 Infilling for Synthetic Data

In the experiment (A), we gathered all sequences without missing symbols, the 513 resulting from the processing of series D, the 725 samples resulting from the augmentation techniques, and the 1,327 duplicates. Afterwards, we randomly removed syllables from the 513 sequences (neither from the augmented or from the duplicated) in order to test the prediction capability of the model. The missing symbols may appear in any position including the beginning and the end of the sequence. In the dataset of experiment (B) (sequences with real gaps), 82.76% of the gaps are in the beginning of the sequence, 12.41% in the middle words, and 4.83% in the end of the sequence. We have tried to replicate that setting in our synthetic experiment. The gaps are distributed as follows over the sequences with synthetic gaps: in the first word 84.09% of the gaps, in the middle words 9.67%, and in the final one 6.24%.

To evaluate the different models on unseen data, considering the scarcity of data, a variation of Leave-One-Out Cross Validation procedure was followed, the *Leave-"one and its derivatives"-Out Cross Validation*, where the derivatives are the samples/sequences resulting from applying an augmentation step (as described in Section 4)

to the real sample/sequence that is currently left out for testing. So, the samples resulting from augmentation of the current test sample sequences were excluded from the respective training set to avoid affecting the test results by including sample sequences of the same origin (through augmentation). Furthermore, only the original sequences were used for testing and not the augmented or the duplicated ones, in order to obtain results that are comparable to those of the experiment with non-augmented data. Thus, the machine learning model is trained 513 times and the final performance estimate is based on all these runs.

We experimented with various models. The first one was with the baseline n-gram method using 2-grams and 3-grams. The n-gram model is a probabilistic LM that estimates the probability of appearance of the nth token given the last n-1 tokens (past context). It is a Markov model of order n-1. Similarly, it can be used to estimate the probability that the next n-1 tokens begin with the current token (future context). Apart from using past and future context separately, we also combined both past and future context within the sequences. The applicability of n-grams is known to be rather limited; for small n, the context used for prediction is also small, while for large n, most test sequences of size n are never seen in the training set, i.e., the method does not scale well for large n; therefore we just used n = 2,3. The results are given in Table 2.

In our following experiments, we used two regular/unidirectional RNNs, a forward-directional RNN which captured information from earlier in the sequence and a backward-directional RNN which captured information from later in the sequence; finally we used a bidirectional RNN, as described in Section 5. We implemented a function that performs one step of stochastic gradient descent with gradient clipping. We applied the greedy heuristic approach to search for the best hyperparameters, ending up with: $epochs = 60$, number of neurons in hidden layer $N_{hl} = 50$, and learning rate $l_r = 0.01$.

At this point it is important to refer to the choice of activation functions for the RNNs. As already stated, as an output activation function, which defines the type of predictions, is set the softmax function, since we are dealing with a d-way classification problem. Regarding the choice of activation function for the hidden layer, which controls how well the network model learns the training dataset, for use both Rectified Linear Activation (ReLU) and Hyperbolic Tangent (Tanh) functions were considered. Comparing the results of linear and nonlinear activation function used for hidden layers, better performance is obtained by Tanh. This happens firstly because we do not have too many layers in our architecture and secondly because our model does not suffer much from the vanishing gradient problem since the Mycenaean sequences are of rather limited size. The choice of initialization of weight matrices and bias vectors was also made through various tests. Zero weight initialization as well as weight-values drawn from a normal and uniform distribution were the first to be tested. *"Glorot/Xavier"* and *"normalized Glorot/Xavier"* initializations were used with Tanh activation function. These two versions of weight initialization are considered the current standard approach of neural network layers and nodes that use the Sigmoid or Tanh activation function [61]. *"He initialization"* of weights was tried with Relu activation function. This version was developed specifically for nodes and layers that use ReLU activation function [62]. We ended up in *"Glorot/Xavier"* [63] as the most suitable for use. Given those choices we came to the results for the RNNs shown in Table 2 (left part) as well.

To demonstrate the benefits of the augmentation step, we present in Table 2 (right part) the results of all the used methods applied only to the original 513 sequences, without augmenting the data.

By considering the results in Table 2, we can infer the following:

— The bidirectional RNN trained with augmented data outperforms the rest of the models especially in TOP-1 and TOP-5 accuracy and is close to optimal for the rest.
— The augmentation appears to help increase TOP-1 and TOP-5 accuracy for the bidirectional RNN and the backward RNN.
— The models utilizing future context/backward states are better than the ones using past context/forward states. This happens most probably due to the high number of missing symbols at the beginning of the tablets; in those cases it is easier to predict backwards.

Table 2. Estimated Scores (Percentages) of Finding the Correct Missing Symbol among the Top-k Most Likely Symbols (k = 1, 5, 10, 15, 20) According to the Probabilities Estimated by the Respective Models

	ORIGINAL AND AUGMENTED DATA					ORIGINAL DATA ONLY				
method	**TOP-1**	**TOP-5**	**TOP-10**	**TOP-15**	**TOP-20**	**TOP-1**	**TOP-5**	**TOP-10**	**TOP-15**	**TOP-20**
2-gram (past context)	17.74	52.63	62.57	69.59	73.29	18.71	52.63	62.38	69.98	73.10
3-gram (past context)	33.33	53.02	57.89	60.23	61.40	33.72	53.02	57.89	59.65	60.82
2-gram (future context)	14.23	53.61	68.62	77.58	**78.95**	14.42	53.22	70.18	77.58	**78.95**
3-gram (future context)	34.50	59.06	67.25	69.20	70.96	34.89	58.28	66.28	68.62	70.57
2-gram (bidirectional)	31.58	64.52	**72.32**	**74.85**	75.24	30.99	64.72	**72.71**	**74.85**	75.24
3-gram (bidirectional)	44.64	56.34	59.45	60.04	60.62	44.44	55.36	58.67	59.26	59.84
RNN (forward states)	21.44	42.88	55.50	64.52	69.52	22.03	42.69	56.14	64.72	70.37
RNN (backward states)	28.27	53.02	64.91	72.32	76.61	27.10	50.49	61.79	69.79	74.07
RNN (bidirectional) - BRNN	**48.34**	**65.30**	71.93	**74.85**	78.17	46.20	60.43	69.01	74.27	77.97
LSTM (bidirectional)	19.49	33.92	48.93	56.14	61.79	15.79	32.36	46.39	55.95	66.08
GRU (bidirectional)	23.97	38.40	53.41	60.62	66.28	17.74	33.14	46.78	54.39	62.38

In the left half of the table we demonstrate the results when both the original and the augmented data were used for training, while in the right half we present the results when only the original data were used.
The bold font indicates the best performance among the methods.

— The models that use both directions are typically better than the unidirectional ones, probably due to the fact that they can better capture the context in both directions.
— The 3-grams outperform 2-grams in TOP-1 accuracy, but perform worse in TOP-10, TOP-15, and TOP-20 accuracy. The 3-grams are more capable to capture the context that gives better TOP-1 predictions, but suffer from singletons (about 62% of 3-grams are singletons), which seem to deteriorate the other results. On the other hand, about 52% of 2-grams are singletons.
— The bidirectional 2-gram suffers from low TOP-1 accuracy but performs rather well for the rest.

To have a complete picture, we also implemented and tested both LSTM and GRU networks (using PyTorch) using a single hidden layer as for the RNN case. The last two rows of Table 2 show the results of the bidirectional LSTM and GRU. The numbers reveal that both networks do not come close to the success of the corresponding RNN model, obviously due to overfitting. However the GRU network provides indications for its future use in experiments with more data, which will result from the processing of other series as well.

Additionally to the above, we evaluated our model assuming that gaps are placed in all possible positions of a sequence using a uniform distribution. The results for TOP-1/5/10/15/20 accuracy using the proposed RNN architecture are (59.87%, 70.19%, 73.88%, 75.48%, 76.68%). Obviously infilling with a uniform distribution of gaps is an easier problem to solve, but does not follow the real gap distribution. Therefore we did not elaborate further in that direction.

This analysis demonstrates that we are able to predict up to a certain extent the missing symbols. It also verifies that the Mycenaean tablets give indications of grammatical inflection and do contain information about the language structure [2].

The results from the augmented dataset/corpus encourage the future study of damaged tablets belonging to smaller series, like, for example, the series K (24 Mycenaean tablets) which records vessels and series R (30 Mycenaean tablets) which records weapons.

6.2 Infilling for Real Cases

In the experiment (B), we aim to verify the applicability of the proposed method in some real cases. To this end, we used the bidirectional RNN model from the previous experiment, which was trained on the whole dataset, including the samples resulting from Section 4, since this is the model that gave the best results so far.

We included all the sequences without missing symbols into the training set, as well as the augmented and the duplicated samples. A total of 145 incomplete sequences (sequences with missing parts), coming from the

Table 3. BRNN Predictions on Representative Real Cases, which Include Missing Symbols at the Beginning, in the Middle, and at the End of the Sequence

Damaged Tablets	Sequences	BRNN TOP-5
KN Dq 46 + fr.	*-sa-no e-ko-so pe-ri-qo-ta-o	*'to', 'ra', 'ka', 'ki', 'ri'*
KN Da 1323 + 5243 + 5325 + fr.	ke-ti-ro e-ra a-no-qo-* OVIS:m	*'jo', 'ta', 'we', 'ja', 'ro'*
KN Dv 1439	ru-na-so qa-* te-ra-po-si-jo	*'ra', 'mo', 'so', 'ro', 'wi'*
KN Dv 1621 + 5116	tu-ti e-* pe-ko OVIS	*'ko', 'ra', 'me', 'ki', 'to'*

damaged tablets of series D (shown in the 4 volumes of the publication *"Corpus of Mycenaean Inscriptions of Knossos"* [20]), were used as our test set. We provide some indicative results in Table 3. In the first column of Table 3, we present four typical cases of damaged tablets, the second column shows the sequences resulting from those tablets of the first column and the third column demonstrates the TOP-5 predictions of the BRNN model. These predictions are sorted using the BRNN output probability in descending order, i.e., the first one gives the most probable syllable according to our model. The full set of estimated syllables on series D is given in the Appendix.

The infillings proposed by the BRNN system for these representative four real cases (see Table 3), namely incomplete sequences resulting from damaged Mycenaean tablets, are commented below:

— On Mycenaean tablet KN Dq 46, it is almost certain that the missing word "*-sa-no" belongs to a person's name. In the Mycenaean bibliography (namely in all series, not only in D), the following names appear with this ending: "**ka**-sa-no" (KN V 831), "a-**ka**-sa-no" (PY Jn 415 and KN As 602 + 650 + 1639 + fr.), "**to**-sa-no" (PY Fn 79), and "**ja**-sa-no"(KN As 1516). Our model also suggests two of them, "to" and "ka".
— Mycenaean tablet KN Da 1323 + 5243 + 5325 + fr. is worn in its center. It is one of the few tablets for which we know with great certainty the missing syllable. It is the syllable "ta" and it is about the name "a-no-qo-**ta**" in the nominative or "a-no-qo-**ta**-o" in the genitive, which belongs to the owner of the herd in the specific area. The syllable "ta" is the second choice of the BRNN system. The names of the owners are more often mentioned in the category D since an owner can be associated with many locations, usually adjacent to each other.
— In the Mycenaean tablet KN Dv 1439, the name of the area (toponym) is ignored. "qa-**ra**" and "qa-**mo**" are the most likely areas, perhaps the only ones that start with the syllable "qa". Both are suggested by the BRNN system and in fact with "qa-**ra**" being more likely than "qa-**mo**".
— The Mycenaean tablet KN Dv 1621 + 5116 is worn at the point where the place-name is written. The most obvious answers are the areas "e-**ko**-so" and "e-**ra**". Both syllables, "ko" and "ra", are included in the suggestions of the BRNN system.

Analyzing the BRNN predictions for the damaged Mycenaean tablets in Table 3, it is demonstrated:

(a) The small size of the Mycenaean sequences. A Mycenaean tablet may offer, for example, only a sequence of two words, thus confirming their brief nature.
(b) The rare repetition of words, mainly human names, confirming that these are aggregate documents of economic-administrative purposes.
(c) The importance of the restoration of the Mycenaean tablets. By contributing to the completion of the Mycenaean sequences, we actually contribute to the enrichment of the Mycenaean lexicon. Most of the words we are asked to fill in are unique.

Then, we used the same model to provide our predictions on some sequences for which there are some expert opinions on the missing parts in the related literature, namely [20]. There are 15 such cases as presented in Table 4. The bibliographic comments on the missing symbol are shown in the third column, and the results of the model in the fourth one. The symbol *"bl"* corresponds to the space symbol.

Table 4. Bibliographic Annotations [20] Comparing to BRNN Predictions in All Sequences Appear in the Real Cases

Damaged Tablets	Sequences	Bibliographic annotation	BRNN TOP-5
KN Dq 447	*-ta-wo da-mi-ni-jo OVIS:m oOVIS:m	*possibly 'ka' or 'qe'*	*'ri', 'ka', 'u', 'ti', 'pi'*
KN Dl 933 + 968 + 975 (Figure 6)	*-*83-re-to si-ja-du-we po-ti-ni-ja-we-jo OVIS:f LANA oLANA oOVISf okiOVIS:m	*perhaps 'ko'*	*'da', 'e', 'po', 'ma', 'ovis:m'*
KN Dp 1061	*-sa pa-i-ti-ja OVIS LANA	*probably 'to'*	*'to', 'ka', 'je', 'qa', '*86'*
KN Dv 1213	*-so u-ta-no OVIS:m	*'to', 'jo' possible*	*'ko', 'ta', 'de', 'pu', 'wo'*
KN Da 1341 [+] 1454 + 8777	*-no-qa-ta pa-i-to da-mi-ni-jo OVIS:m	*'po' possible, but difficult*	*'wo', 'jo', 'ta', 'ku', 'o'*
KN Db 1344 + 6017 + 7268 + 7950 + 8235	*-tu-to pa-i-to we-we-si-jo-jo OVIS:m OVIS:f	*perhaps 'ti'*	*'ti', 'bl', 'ke', 'ra', 'ka'*
KN Da 1401 + 7998 + fr.	*-wi da-*22-to OVIS:m	*perhaps 'na'*	*'ti', 'ri', 'to', 'na', '*22'*
KN Df 5198 + 5238 + 5269	wi-na-jo ra-* ki-ri-jo-te OVIS:m OVIS:f peOVIS:m	*traces favour 'ja'*	*'ja', 'to', 'bl', 'jo', 'ko'*
KN Dv 5236 + 5329 (Figure 7)	*-jo ra-to OVIS:m	*perhaps 'qa' or 'wo'*	*'ri', '*18', 'ti', 'te', 'na'*
KN Dv 5278 + 5338 + 8557	*-ma-we qa-mo OVIS:m	*'ko' not impossible*	*'ra', 'ko', 'ja', 'ri', '*56'*
KN Db 5310 + 6062 + 8375 (Figure 8)	e-*-jo ku-ta-to OVIS:m OVIS:f	*perhaps 'qa' or 'ri'*	*'ko', 'ki', 'da', 'ka', 'wi'*
KN Db 5359 + 5565 + 7214	*-ma-na-so ra-su-to u-ta-jo OVIS:m OVIS:f	*'pi' not impossible*	*'bl', 'ta', 'ri', 'mi', 'su'*
KN Dv 5690	du-ni-*	*'jo' possible*	*'ja', 'bl', 'wa', 'ma', 'u'*
KN Dc 7161 + 7179 + 8365 + fr. (Figure 9)	*-to ku-ta-to u-ta-jo-jo OVIS:m oOVIS:m	*possibly 'ke'*	*'ra', 'me', 'ka', 'a2', 'a3'*
KN Do 7740	*-ta ka-to-ro se-to-i-ja	*'ke' or 'de'*	*'ri', 'ti', 'si', 'me', 'nu'*

The experts base their guesses mainly on the visual cues, since some small parts of the syllables remain visible, and not on the sequences' structure, unlike our method. For example, in the damaged tablet of Figure 6, it is estimated that the remnant of the missing syllable matches in appearance/schematically with the Mycenaean syllable *"ko"*. On the other hand, we make our prediction based on the grammatical rules of the language.

Six of our TOP-5 predictions agree with the literature recommendations. For cases where there is no identification between the TOP-5 predictions and the bibliographic annotations, the BRNN predictions were also

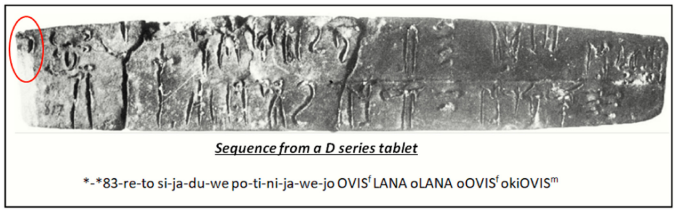


Fig. 6. The image of the damaged Mycenaean tablet KN Dl 933 + 968 + 975 on its left side (image copyright belongs to the Hellenic Ministry of Culture and Sports - "Hellenic Organization of Cultural Resources Development"). Translation of the Mycenaean tablet *"*-*83-re-to (incomplete shepherd's name): at si-ja-du-we (place name) belonging to po-ti-ni-ja-we-jo (Potniaweios), rams, wool, deficit wool ('o' is abbreviation of the Greek word 'ophelos'), deficit rams and deficit ki-rams"*.

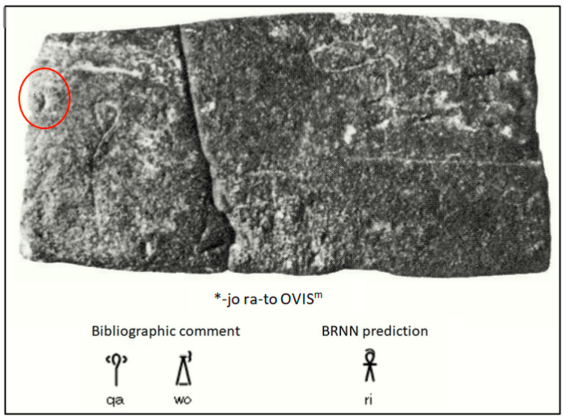


Fig. 7. The image of the damaged Mycenaean tablet KN Dv 5236 + 5329 (copyright belongs to the Hellenic Ministry of Culture and Sports - "Hellenic Organization of Cultural Resources Development") and its proposed supplement from the bibliography [20] (left) and from the BRNN model (right). Translation of the Mycenaean tablet *"(unknown shepherd's name): at ra-to (place name), rams"*.

checked visually whether they match the remnants on each tablet. For example, in the tablets KN Dv 5236 + 532 and KN Db 5310 + 6062 + 837 (see Figures 7 and 8), we notice that the predictions of our model, and in fact the first, are not completely irrelevant/unrelated to the remnant. The same goes for the tablet KN Dv 5690 where the bibliography suggests the syllable 'jo', *du-ni-jo*, but also the prediction of the model, 'ja', cannot be ignored since it matches the remnant and also gives a human/personal name, *du-ni-ja*, probably female. Tablet KN Dc 7161 + 7179 + 8365 + fr. (Figure 9) is worn on its left side, and in fact in the first syllable of the two-syllable word. The bibliography suggests the word *'ke-to'*, but the proposal of our model, *'a3-to'* cannot pass unnoticed. Both human names appear in series D and the syllable 'a3' may be more in line with the handwriting of the scribe. In the remaining five tablets, there is no clear correlation between the model's predictions and the remnant. In the following, we try to give some possible explanations, since there is no hard evidence available:

— On the tablet Dl 933 (Figure 6), we find a syllable, '*83' (its phonetic value has not been determined with certainty), which is quite rare and is observed only twice in the training dataset. As a result, the model is probably not sufficiently trained in such a context.
— The TOP-5 BRNN predictions are not consistent with the remnant in the KN Dv 1213. The model prioritizes other syllables, probably due to the short sequence length, which conveys rather poor context information. Inclusion of visual evidence in our model in the future could handle such issues.

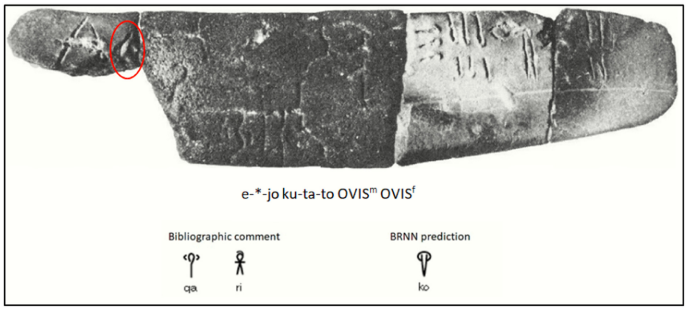


Fig. 8. The image of the damaged Mycenaean tablet KN Db 5310 + 6062 + 8375 (copyright belongs to the Hellenic Ministry of Culture and Sports - "Hellenic Organization of Cultural Resources Development") and its proposed supplement from the bibliography [20] (left) and from the BRNN model (right). Translation of the Mycenaean tablet *"(damaged shepherd's name e-*-jo): at ku-ta-to (Kutaiton, place name), rams and ewes"*.

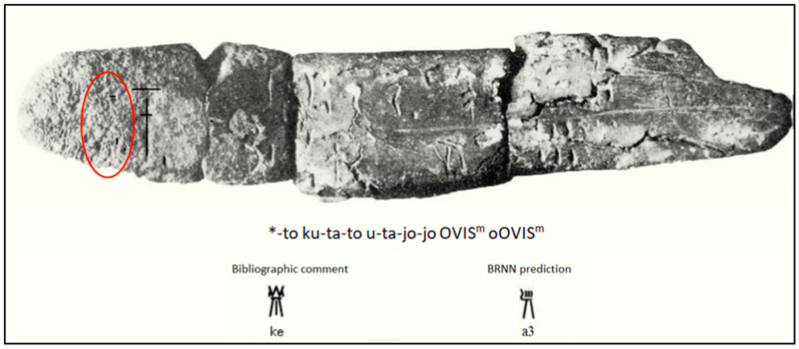


Fig. 9. The image of the damaged Mycenaean tablet KN Dc 7161 + 7179 + 8365 + fr. (copyright belongs to the Hellenic Ministry of Culture and Sports - "Hellenic Organization of Cultural Resources Development") and its proposed supplement from the bibliography [20] (left) and from the BRNN model (right). Translation of the Mycenaean tablet *"(unknown shepherd's name): at ku-ta-to (Kutaiton, place name) belonging to u-ta-jo (collector's name), rams and deficit rams"*.

— The tablet Da 1341 is a difficult case, since the visual evidence is very weak and even the experts note that their degree of certainty is low.
— The tablet Db 5359 + 5565 +7214 is another difficult case as acknowledged by experts and their degree of certainty is also low.
— The remnant in tablet KN Do 7740 largely matches the syllable 'ke', suggested by the bibliography. The model in this case probably failed to predict part of a human name; human names are typically unique.

The visual search of the missing symbol becomes more complicated due to the fact that the handwriting may differ from the syllabary that we use as reference. It is important to note that the tablets in series D were written mostly by one scribe and so the problem of dealing with different handwritings is limited, but still the scribe's handwriting can deviate from the reference.

Clearly our method provides a prior on sequence structure, while the visual appearance of the missing part gives a visual observation. These are complementary sources of information which can be combined by experts to infill the missing parts. Our model can serve as a tool for the expert helping him/her to choose from the various plausible alternatives (e.g., from TOP-5).

7 CONCLUSIONS AND FUTURE WORK

We presented a generative neural LM for the most ancient proven stage of the Greek language, the Mycenaean Greek language attributed by Linear B script. We collected a dataset of sequences extracted from partially damaged Mycenaean documents in the related literature. We demonstrated that it is possible to train our LM in order to infill the damaged parts of Mycenaean inscriptions. This is the first such effort to our best knowledge.

Our model exploits a symbol-level BRNN in order to capture the statistical structure of the Mycenaean documents. We verified our method experimentally using synthetic and real data, we compared to experts' opinions and suggested possible explanations in cases of discrepancies between them. Our methodology is expected to assist the experts recover the missing parts by offering alternatives along with their probability, which are complementary to the visual channel.

The key takeaways are as follows:

— The rather limited amount of data does not allow us to make full use of state-of-the-art methods for language modeling; however the classic RNN seems to be able to capture the context and provide results that are better than n-gram models, which are often used in such applications.
— The data augmentation can help get better results. However, the augmentation step is not trivial at all. It requires deep knowledge of the language and the domain and has to be done with caution in order to avoid contamination of the dataset with misleading samples.
— The generative models like ours can provide alternatives for infilling the damaged parts, along with their probabilities based solely on the structure of the documents discovered so far. Our model can be a complementary tool for the experts, who as stated in the literature have based their estimates mainly on the visual appearance of the missing parts. That appearance could be sometimes misleading or, in many cases, not available at all.

We hope that this work will help bring closer the machine learning, the linguistic and the archaeological communities and will spark the interest of analyzing the Mycenaean Linear B and other ancient scripts, either deciphered or undeciphered, using the appropriate sequential modeling tools.

In the future, we will also seek for new ways to transfer the acquired knowledge from series D to the other Linear B series and vice versa. We will investigate new data augmentation rules that seem necessary in tasks that have to manage a small amount of data. Finally the visual channel could be considered, i.e., the partially visible symbols, could help define constraints to scale down the feasible solution space.

APPENDIX

A MODEL OUTPUT FOR DAMAGED TABLETS OF SERIES D

Table 5. BRNN Predictions in the Remaining 127 Real Cases

Damaged Tablets	**Sequences**	**BRNN TOP-5**
KN Dq 42	*-jo ma-sa pe-ri-qo-ta-o OVIS:m	*'ri', 'ni', 'te', 'ti', 'si'*
KN Dp 43	*-ta-re-wo OVIS:m LANA	*'u', 'ku', 'we', 'lana', 'me'*
KN Dq 45	*-mo e-ra a-no-qo-ta-o OVIS:m	*'so', 'ri', 'je', 'ni', 'te'*
KN Dl 412	*-da ka-ru-no sa-qa-re-jo OVIS:f okiOVIS:m	*'si', 'u', 'to', 'pe', 'pi'*
KN Dq 438	*-to-i-ja ka-to-ro se-to-i-ja OVIS:m	*'ki', 'se', 'ke', 'da', 'po'*
KN Dq 440	*-ka-mo a-no-qo-ta e-ra OVIS:m oOVIS:m	*'ku', 'a', 'ta', 'po', 'pa'*
KN Dq 441	*-qo-si-jo o-re-te-wo da-*22-to	*'ra', 'ri', 'ta', 'ku', 'we'*
KN Dq 448	*-ni-to u-ta-no	*'mi', 'ti', 'du', 'pa', 'bl'*
KN Dk 920 + 7294 + 7330	*-ni-ja-so da-*22-to ko-ma-we-to OVIS:m LANA oLANA	*'ta', '*82', 'tu', 'mi', 'du'*
KN Dl 928	*-*56-na-ro da-wo ra-wo-qo-no-jo OVIS:f kiOVIS	*'wi', 'jo', 'ru', 'wa', 'ra'*
KN Dk 931 + 7293	*-da sa-jo ko-ma-we-to OVIS:m LANA	*'i', '*22', 'te', 'de', 'ra'*
KN Dl 934 + 7082	*-ja e-ko-so OVIS:f oOVIS:f WE kiOVIS:m okiOVIS:m	*'ni', 'ti', 'pa', 'te', 'ra'*
KN Dl 935 + 942	*-nu-ka da-ta-ra-mo sa-qa-re-jo OVIS:f LANA oLANA oOVIS:f okiOVIS:m	*'du', 'ni', 'da', 'ti', 'ki'*
KN Dl 938	*-re-jo OVIS:f kiOVIS:m LANA okiOVIS:m oLANA	*'pa', 'a', 'qa', 'mi', 'mo'*
KN Dl 940 + 8779	*-so e-ra sa-qa-re-jo OVIS:f LANA okiOVIS:m LANA	*'de', 'pu', '*22', 'je', 'nwa'*
KN Dk 969	*-ki-to a-ko-ra si-ja-du-we	*'ru', 'bl', 'ra', 'ko', 'ni'*
KN D 1024 + 7130	*-ra-jo ma-ti-jo OVIS	*'u', 'wa', 'e', 'qa', 'we'*
KN Dk 1066	*-te-u ku-ta-to kineOVIS:m LANA	*'ke', 'ka', 'qo', 'ko', 'te'*
KN Dk 1068	*-*49-so ku-ta-to OVIS:m LANA oLANA	*'ri', 'ta', 'mi', 'pu', 'lana'*
KN Dk 1070	*-za-ra-ro ku-ta-to OVIS:m LANA oLANA	*'i', 'za', 'ma', 'de', 'mi'*
KN Db 1086	*-ro da-*83-ja a-ka-ta-jo OVIS:m OVIS:f	*'ko', 'to', 'ru', 'za', 'pu'*
KN Da 1087	*-jo a-ka OVIS:m	*'ri', 'te', 'wi', 'ti', 'ni'*
KN Da 1091 + 1413	*-ri-sa-ta ri-jo-no OVIS:m	*'ti', 'pe', 'nu', 'qe', 'te'*
KN Da 1091 + 1413 (verso)	su-ki-ri-ta *-nu-ra-qa	*'ovis:m', 'e', 'ovis:f', 'neovis:m', 'saovis'*
KN Da 1091 + 1413 (verso)	na-u-ti ka-ta-do-qe *	*'ovis:f', 'peovis:m', 'ovis', 'ovis:m', 'se'*
KN De 1109	*-te-to ku-ta-to u-ta-jo OVIS:m OVIS:f oOVIS:m	*'wo', 'ta', 'ka', 're', 'ke'*
KN Dv 1124	*-ta ku-ta-to OVIS:m	*'pe', 'we', 'u', 'lana', 'a3'*
KN Db 1126 + 5303 + 7208	*-no-ta-na ku-ta-to OVIS:m OVIS:f	*'jo', 'o', 'ja', 'i', 'wi'*
KN De 1141	*-ra u-ta-jo-jo OVIS:m OVIS:f oOVIS:m	*'e', 'qa', 'da', 'we', 'wi'*
KN Dv 1142	*-jo-ko e-ko-so u-ta-jo-jo OVIS:m	*'ri', 'di', 'ki', 'ti', 'ra'*
KN Dv 1145	*-so da-wo u-ta-jo-jo OVIS:m OVIS:f	*'pu', 'de', 'ko', 'ta', 'ru'*
KN Dv 1146 + 1498	*-mo-ni-ja-ro da-wo u-ta-jo-jo OVIS:m OVIS:f	*'bl', 'qa', 'so', 'ru', 'no'*
KN Dc 1148	*-ma-ro da-*22-to OVIS:m peOVIS:m	*'ri', 'te', 'ke', 'ti', 'ta'*
KN Db 1165 + 7110 + 7226	*-ra-ko da-ra-ko we-we-si-jo OVIS:m OVIS:f	*'di', 'mi', 'da', 'we', 'pu'*
KN Df 1187 + 7191	*-ri-jo ra-to OVIS:m OVIS:f peOVIS:m	*'au', 'ti', 'ki', 'mi', 'we'*
KN Dd 1193 + 5370	*-du-ro ra-to OVIS:m OVIS:f paOVIS:m	*'a', 'te', 'ke', 'ti', 'ri'*
KN Da 1194	*-de-u ra-su-to ki-ri-jo-te OVIS:m	*'ke', 'te', 'ta', 'ti', 'ra'*
KN Db 1196 + 8233	*-pa-u-ro ra-su-to OVIS:m OVIS:f	*'ke', 'ma', 'te', 'de', 'zo'*
KN Db 1208 + 5488	*-ra-so ri-jo-no OVIS:m OVIS:f	*'e', 'qa', 'wa', 'o', 'du'*
KN Db 1211 + 5389	*-ru-ka-*18 ri-jo-no OVIS:m OVIS:f	*'de', 'ko', 'ma', '*18', 'bl'*
KN Dh 1243	*-e-to tu-ri-so kiOVIS:m	*'ra', 'bl', 'ke', 're', 'se*
KN Da 1273 + 1440 [+] 8788	*-ne-u e-ko-so OVIS:m	*'te', 'ra', 'ta', 'i', 'ke'*
KN Dv 1309	*-wo-no do-ti-ja a-te-jo	*'ta', 'ka', 'ri', 'ti', 'di*

(Continued)

Table 5. Continued

Damaged Tablets	Sequences	BRNN TOP-5
KN Dg 1318	*-ta-ro qa-mo OVIS:m paOVIS:m oOVIS:m	*'ri', 'pe', 'ka', 'lana', 'ti'*
KN Dv 1328 + frr.	*-*49-so ku-ta-to OVIS:m OVIS:m	*'ta', 'ri', 'mi', 'pu', 'ka'*
KN Dv 1334 + 5324 + 8393 + frr.	*-ja pe-ri-qo-te-jo ki-ri-jo-te OVIS:m	*'ni', 'ra', 'ti', 'we', 'i'*
KN Db 1340 + 5263 + fr.	*-ra-ro su-ri-mo OVIS:m OVIS:f	*'da', 'qa', 'ta', 'we', 'pu2'*
KN Dd 1342	pa-i-to *-ra-to-jo OVIS:f OVIS:m paOVIS:m	*'we', 'qa', 'pa', 'ma', 'se'*
KN Da 1352 + 5634 + frr.	*-ti-jo pa-i-to u-ta-jo-jo OVIS:m	*'wo', 'do', 'za', 'mi', 'pa'*
KN Dd 1366 + 5360	*-na da-wo OVIS:m OVIS:f paOVIS:m	*'ru', 'ka', 'u', 'ro', 'o'*
KN Dq 1377 + frr.	pa-*-na ti-ri-to OVIS:m	*'i', 'o', 'e', 'wi', 'pu'*
KN Da 1382 + 1482	*-mi-ni-to ri-jo-no u-ta-jo OVIS:m	*'ti', 'i', 'pi', 'te', 'a3'*
KN De 1383 + 5679	*-ro ku-ta-to u-ta-jo-jo OVIS:m OVIS:f oOVIS:m	*'qi', 'qa', 'wi', 'mi', 'za'*
KN Dv 1386 + 8575	ti-ri-to qa-* a-te-jo	*'ra', 'mo', 'sa', 'no', 'okiovis'*
KN Dk 1399	*-se-u da-mi-ni-jo OVIS:m	*'te', 'ke', 'ti', 'ri', 'ka'*
KN Dv 1411	*-ri-ja-ta	*'ti', '*56', 'au', 'pe', 'ki'*
KN Dv 1422	*56-ro2 su-* OVIS:m	*'ri', 'to', 'ma', 'ja', 'ka'*
KN Dd 1429 + 5264 + 5327	*-za-ra-ro pa-i-to u-ta-jo OVIS:m OVIS:f paOVIS:m	*'wo', 'za', 'mi', 'ra', 'ta'*
KN Dv 1430 + 7200 + 8410 + fr.	*-do-ro da-ra-ko we-we-si-jo OVIS:m OVIS:f OVIS:m	*'ku', 'ti', 'te', 'da', 'ta'*
KN Dv 1501	di-*79-nu-*	*'u', 'me', 'ne', 'da', 'ra'*
KN Dv 1504 + fr.	di-de-ro e-* OVIS:m OVIS:f	*'ra', 'ko', 'mo', 'ja', 'me'*
KN Dv 1506	di-za-*	*'nu', 'ti', 'na', 'da', 'si'*
KN Dd 1579 + 1586 + fr.	*-ti da-ra-ko we-we-si-jo OVIS:m OVIS:f paOVIS:m	*'za', 'mi', 'pu', '*22', 'do'*
KN Dv 1607 + 5978 + 7276 + fr.	*-ti-ro pa-i-to we-we-si-jo OVIS:m OVIS:f	*'do', 'ti', 'za', 'ri', 'mi'*
KN D 1650	*-pa da-wo ra-wo-qo-no-jo OVIS:m	*'ra', 'pe', 'we', 'u', 'ta'*
KN Da 2005 + 5366	*-ro a-ka OVIS:m	*'pe', 'po', 'ra2', 'za', 'te'*
KN Db 2020 + 5314 [+] 5423	*-ru-wo-i-ko su-ri-mo OVIS:m OVIS:f	*'te', 'bl', 'na', 'po', 'za'*
KN Da 2027 + fr.	*-*22 ku-ta-to OVIS:m	*'da', 'za', 'olana', 'wo', 'wi'*
KN Db 5041 + 8382 + fr.	*-ni tu-ni-ja OVIS:m OVIS:f	*'me', 'ti', '*86', 'mo', 'du'*
KN Dv 5075	*-mo pa-i-to we-we-si-jo OVIS:m oOVIS:m	*'te', 'ri', 'je', 'so', 'nu'*
KN Dd 5174 + 5215	*-da-na-ro pa-i-to OVIS:m OVIS:f paOVIS:m	*'bl', 'wi', 'jo', '*82', 'de'*
KN Da 5179 + 5674 + 7257 + 8556 + 8611	*-ti-ko-ro ku-ta-to OVIS:m	*'*22', 'za', 'wo', 'do', 'i'*
KN Dc 5190 + 7157 + 8194	*-de-a-ta *56-ko-we OVIS:m peOVIS:m	*'ta', 'wo', 'me', 'ne', 'we'*
KN Da 5209 + 5248	*-jo pu-so OVIS:m	*'te', 'we', 'ti', 'pi', 'ni'*
KN Da 5220 + 5330 + 5355	*-ro ku-ta-to OVIS:m	*'qi', 'qa', 'wi', 'mi', 'za'*
KN Dv 5224 + fr.	*-a-pa-ni-jo OVIS:m	*'bl', 'ti', 'to', 'ki', 'zaovis:m'*
KN Da 5245 + 5299	*-ro e-ko-so ki-ri-jo-te OVIS:m	*'da', 'to', 'ze', 'je', 'ti'*
KN Db 5272 [+] 5294	ru-ta2 ra-* ki-ri-jo-te OVIS:m OVIS:f	*'ja', 'to', 'wi', 'jo', 'po'*
KN Df 5275 + 5581 + 7078	*-ti ri-jo-no OVIS:m OVIS:f peOVIS:m	*'za', 'e', 'qe', 'a2', 'mu'*
KN Dn 5286 + 5362	*-ka-ta OVIS:m	*'ku', 'a', 'po', 'da', 'du'*
KN Dv 5291 + 5588	*-ri-ro su-ri-mo OVIS:m	*'ti', '*56', 'ki', 'pe', 'di'*
KN Dv 5297	*-ti-jo qa-mo OVIS:m	*'wo', 'do', 'za', 'ko', 'mi'*
KN Da 5308 + 5332 + 7694	si-*-te-u ku-ta-to OVIS:m	*'ka', 'ta', 'o', 'ke', 'da'*
KN Dv 5312 [+] 6057 [+] 8789	wo-jo-* e-ko-so OVIS:m	*'jo', 'no', 're', 'ro', 'qo'*
KN Dv 5328 + 5376	*-ro ku-ta-to OVIS:m	*'qi', 'qa', 'wi', 'mi', 'za'*
KN De 5335 + 9660 + frr.	*-ri ku-ta-to oOVIS:m	*'du', 'ti', 'au', 'za', 'pe'*
KN Dv 5346	ta-pa-*	*'i', 'bl', 're', 'si', 'ja'*
KN Dv 5349	*-nwa pu-so	*'we', 'qo', '*86', 'ka', 'da'*
KN Dv 5398 + 7207 + 8749 + fr.	*-no ku-ta-to OVIS:m OVIS:f	*'ko', 'wi', 'mo', 'da', 'qi'*
KN Df 5406 + 8371	*-jo-ro ra-su-to OVIS:m OVIS:f peOVIS:m	*'ri', 'ti', 'ni', '*18', 'te'*
KN Da 5576 + 7160	*-ki-so e-ra u-ta-jo-jo OVIS:m	*'pu', 'ri', 'ru', 'su', 'mi'*
KN Dn 5668 + fr.	*-qo-te-jo OVIS:m	*'ri', '*18', 'wo', 'ra', 'ku'*

(Continued)

Table 5. Continued

Damaged Tablets	Sequences	BRNN TOP-5
KN Dd 5704 + 7180 + frr.	ri-pa-* ra-ja OVIS:m OVIS:f paOVIS:m	*'ko', 're', 'e', 'ja', 'wo'*
KN Db 5715 + 7274 + 7942 + 8374 + fr.	*-ti-jo da-*22-to OVIS:m OVIS:f	*'wo', 'do', 'za', 'ri', 'ko'*
KN Dv 5843 + 8405	*-ti da-wo OVIS:m OVIS:f	*'qe', 'za', 'ru', 'i', 'do'*
KN D 5954	*-ke-mo saOVIS:m	*'su', 'de', 'ovis:m', 'we', 'a3'*
KN Dv 6018 + 8358 + fr.	ke-*-na-ro tu-ni-ja	*'u', 'wa', 'da', 'du', 'pu'*
KN Dv 6045	pu-ma-*	*'da', 'so', 'me', 'di', 'ma'*
KN Dl 7071	*-no ri-jo-no OVIS:f LANA oOVIS:f oLANA	*'wo', 'i', 'wi', 'ku', 'ka'*
KN Da 7080	*-ro *56-ko-we OVIS:m	*'wi', 'da', 'to', 'pe', 'mu'*
KN Da 7081	*-ja ki-ri-jo-te OVIS:m	*'ni', 'ra', 'olana', '*22', 'me'*
KN Do 7087	*-ro se-to-i-ja OVIS:f kiOVIS	*'to', 'ko', 'su', 'pe', 'po'*
KN Db 7118 + 7229 + 7881 + fr.	*-ma qa-na-no-to OVIS:m OVIS:f	*'e', 'i', 'ra', 'me', 'du'*
KN Dl 7125	*-to ri-jo-no OVIS:f oOVIS:f	*'ne', 'ra', 'a3', 'a2', 'ke'*
KN Dq 7126	pa-i-to *-ko-ta-o OVIS:m	*'a', 'u', 'ovis:m', 'kiovis:m', 'wi'*
KN Dv 7140	*-wo e-ra OVIS:m paOVIS:m	*'da', 'je', 'wo', 'te', 'de'*
KN Dl 7141 + 7264 + 7971 + 7984	*-ti-za *56-ko-we-e sa-qa-re-jo okiOVIS:m	*'mi', 'oovis:f', 'qe', 'pu', 'do'*
KN D 7334	*-ni-to e-ra a-no-qo-ta	*'ti', 'mi', 'pa', 'du', 'ma'*
KN D 7386	*-ro u-ka OVIS:m	*'ko', 'to', 'ru', 'ta', 'de'*
KN Dl 7771	*-qo-ta ra-ja po-ti-ni-ja-we-jo	*'ri', 'mo', 'no', 'i', 'ni'*
KN D 7773	*-ta-ra-pi OVIS:m	*'du', 'u', 'mi', 'ri', 'ku'*
KN Dl 7875 + 9714 + 9862	*-ku-mi-ro ra-ja	*'jo', 'o', 'e', 'ta', 'bl'*
KN Dl 7905 + 9328 + 9332 + fr.	*-ra-ri-jo si-ja-du-we po-ti-ni-ja-we-jo	*'bl', 'we', 'da', 'ta', 'u'*
KN Dv 8193	*-su-ti-jo a-ka	*'ra', 'ri', 'bl', 'ko', 'ma'*
KN Da 8228 + fr.	*-nwa-jo da-wo u-ta-jo-jo OVIS:m	*'we', 'ta', 'mi', 'me', '*82'*
KN Dv 8280 + 8710 + 9584 + 9656 + frr.	qa-mi-ki-*	*'to', 'u', 'da', 'ti', 'si'*
KN Da 8355 + fr.	*-ki-si-wo ra-su-to OVIS:m	*'di', 'ta', 'we', 'da', 'pe'*
KN Db 8360 + fr.	*-zo ra-su-to OVIS:m OVIS:f	*'te', 'je', 'e', 'me', 'to'*
KN Dv 8361 + fr.	*-to da-wo OVIS:m OVIS:f	*'ka', 'ne', 'te', 'i', 'ra'*
KN Dv 8366	*-*56-wo su-ri-mo	*'ru', 'ri', 'seovis:f', 'qa', 'bl'*
KN Db 8383 + 8415 + 8426 + 8547	*-ko-so ra-to OVIS:m OVIS:f	*'e', 'ri', 'u', 'pu', 'a'*
KN Dv 8634	*-za-ra-ro	*'ke', 'mi', 'ru', 'ta', 'i'*
KN Dv 8652 + frr.	a-ru-* da-*22-to OVIS:m oOVIS:m	*'*56', 'ro', 'je', 'sa', 'so'*
KN Dl 9016 + 9428 + frr.	*-di-de-ro tu-ni-ja OVIS:f kiOVIS LANA kiOVIS	*'ma', 'bl', 'wi', 'ta', 'ke'*
KN D 9290	*-to-wi-jo OVIS:m	*'ra', 'wo', 'ri', 'we', 'za'*
KN Dl 9841	*-qo-no-jo OVIS:f kiOVIS:m	*'ri', 'sa', 'ra', 'ku', '*18'*

ACKNOWLEDGMENTS

We thank the Heraklion Archaeological Museum and the Hellenic Ministry of Culture and Sports - "Hellenic Organization of Cultural Resources Development (HOCRED)" for providing the images of the inscriptions. We wish to thank Dr Ester Salgarella for the useful discussions and the insightful suggestions.

REFERENCES

[1] J. Chadwick. 1990. *The Decipherment of Linear B.* Cambridge University Press. Retrieved from https://books.google.gr/books?id=KOaMAgAAQBAJ.

[2] Margalit Fox. 2013. *Riddle of the Labyrinth: The Quest to Crack an Ancient Code and the Uncovering of a Lost Civilisation.* Profile Books.

[3] C. Renfrew. 1972. *The Emergence of Civilisation: The Cyclades and the Aegean in the Third Millennium B.C.* Methuen. Retrieved from https://books.google.gr/books?id=B0toAAAAMAAJ.

[4] J. Chadwick. 1976. *The Mycenaean World*. Cambridge University Press, Cambridge/London/NewYork, NY/Melbourn.
[5] E. Vermeule. 1972. *Greece in the Bronze Age*. University of Chicago Press. Retrieved from https://books.google.gr/books?id=92qpNAEACAAJ.
[6] M. Ventris and J. Chadwick. 2015. *Documents in Mycenaean Greek*. Cambridge University Press. Retrieved from https://books.google.gr/books?id=AkgPCAAAQBAJ.
[7] Y. Duhoux and A. M. Davies. 2008. *A Companion to Linear B: Mycenaean Greek Texts and Their World*. Vol. 1–3, Bibliotheque Des Cahiers de L'Institut de Linguistique de Louvain, Peeters Press.
[8] J. T. Hooker. 1980. *Linear B: An Introduction*. Bristol Classical Press.
[9] M. Ventris and J. Chadwick. 1953. Evidence for Greek dialect in the Mycenaean archives. *Journal of Hellenic Studies* 73 (1953), 84–103. Retrieved from https://books.google.gr/books?id=539ktAEACAAJ.
[10] M. Ventris and J. Chadwick. 1956. *Evidence for a Greek Dialect in the Mycenaean Archives*. Council of the Society for the Promotion of Hellenic Studies. Retrieved from https://books.google.gr/books?id=SYHRSgAACAAJ.
[11] J. T. Killen. 1964. The interpretation of Mycenaean Greek texts by L. R. Palmer. Oxford: The Clarendon Press, 1963. 501 pp., frontispiece, 4 figs. 70s. *Antiquity* 38, 150 (1964), 148–150. DOI: http://dx.doi.org/10.1017/S0003598X00030799
[12] Mario Doria. 1965. *Avviamento Allo Studio Del Miceneo: Struttura, Problemi e testi / Mario Doria*. Edizioni dell'Ateneo, Roma.
[13] M. Ventris, A. Sacconi, and J. Chadwick. 1988. *Work Notes on Minoan Language Research and Other Unedited Papers*. Edizioni dell'Ateneo. Retrieved from https://books.google.gr/books?id=h0VoAAAAMAAJ.
[14] Torsten Meissner. 2001. F. M. J. Waanders: Studies in local case relations in Mycenaean Greek. Pp. vii 134. Amsterdam: J. C. Gieben, 1997. Paper, Hfl. 65. ISBN: 90-5063-107-X. *The Classical Review* 51, 1 (2001), 179–180. DOI: http://dx.doi.org/10.1093/cr/51.1.179
[15] A. Robinson and S. Eisenman. 2002. *The Man who Deciphered Linear B: The Story of Michael Ventris*. Thames & Hudson. Retrieved from https://books.google.gr/books?id=SR50QgAACAAJ.
[16] Maurice Pope. 2008. The decipherment of Linear B. In *A Companion to Linear B: Mycenaean Texts and their World*. Anna Morpurgo Davies and Yves Duhoux (Eds.), Vol. 1, Peeters, Louvain-la-Neuve, 3–11.
[17] Y. Duhoux and A. M. Davies. 2008. *A Companion to Linear B: Mycenaean Greek Texts and Their World*. Peeters. Retrieved from https://books.google.gr/books?id=UC8-tAEACAAJ.
[18] M. D. Freo and M. Perna. 2019. *Manuale di Epigrafia Micenea: Introduzione Allo Studio Dei Testi in Lineare B*. Libreriauniversitaria.it edizioni. Retrieved from https://books.google.gr/books?id=mnqKxwEACAAJ.
[19] A. Bernabé and E. R. Luján. 2020. *Introducción al griego micénico. Gramática, selección de textps y glosario*. Prensas de la Universidad de Zaragoza. Retrieved from https://books.google.gr/books?id=QAILEAAAQBAJ.
[20] J. Chadwick, L. Godart, J. T. Killen, J. P. Olivier, A. Sacconi, and I. A. Sakellarakis. 1987. *Corpus of Mycenaean Inscriptions from Knossos: Volumes 1–4*. Cambridge University Press.
[21] Federico Aurora. 2015. DāMOS (Database of Mycenaean at Oslo). Annotating a fragmentarily attested language. *Procedia - Social and Behavioral Sciences* 198 (July 2015), 21–31. DOI: http://dx.doi.org/10.1016/j.sbspro.2015.07.415.
[22] Tomas Mikolov, Martin Karafiát, Lukás Burget, Jan Cernocký, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Proceedings of the 11th Annual Conference of the International Speech Communication Association*. ISCA, 1045–1048.
[23] H. Xu, K. Li, Y. Wang, J. Wang, S. Kang, X. Chen, D. Povey, and S. Khudanpur. 2018. Neural network language modeling with letter-based features and importance sampling. In *Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing*. 6109–6113. DOI: http://dx.doi.org/10.1109/ICASSP.2018.8461704
[24] Yasumasa Miyamoto and Kyunghyun Cho. 2016. Gated word-character recurrent language model. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Austin, TX, 1992–1997. DOI: http://dx.doi.org/10.18653/v1/D16-1209
[25] Krzysztof Wolk and Krzysztof Marasek. 2014. Polish-English speech statistical machine translation systems for the IWSLT 2014. In *Proceedings of the 11th International Workshop on Spoken Language Translation: Evaluation Campaign*. Lake Tahoe, CA, 143–149. Retrieved from https://aclanthology.org/2014.iwslt-evaluation.21.
[26] Ethan Fetaya, Yonatan Lifshitz, Elad Aaron, and Shai Gordin. 2020. Restoration of fragmentary Babylonian texts using recurrent neural networks. *Proceedings of the National Academy of Sciences* 117, 37 (2020), 22743–22751. DOI: http://dx.doi.org/10.1073/pnas.2003794117 arXiv:https://www.pnas.org/content/117/37/22743.full.pdf
[27] Wilson L. Taylor. 1953. "Cloze procedure": A new tool for measuring readability. *Journalism Quarterly* 30, 4 (1953), 415–433. DOI: http://dx.doi.org/10.1177/107769905303000401
[28] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, MN, 4171–4186. DOI: http://dx.doi.org/10.18653/v1/N19-1423
[29] William Fedus, Ian Goodfellow, and Andrew Dai. 2018. MaskGAN: Better text generation via filling in the ____. In *Proceedings of the International Conference on Learning Representations*. Retrieved from https://openreview.net/pdf?id=ByOExmWAb.

[30] Chris Donahue, Mina Lee, and Percy Liang. 2020. Enabling language models to fill in the blanks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 2492–2501. DOI: http://dx.doi.org/10.18653/v1/2020.acl-main.225

[31] Wanrong Zhu, Zhiting Hu, and Eric P. Xing. 2019. Text infilling. arXiv:1901.00158v2. Retrieved from http://arxiv.org/abs/1901.00158.

[32] Dayiheng Liu, Jie Fu, Pengfei Liu, and Jiancheng Lv. 2019. TIGS: An inference algorithm for text infilling with gradient search. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, 4146–4156. DOI: http://dx.doi.org/10.18653/v1/P19-1406

[33] Tianxiao Shen, Victor Quach, Regina Barzilay, and Tommi Jaakkola. 2020. Blank language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Online, 5186–5198. DOI: http://dx.doi.org/10.18653/v1/2020.emnlp-main.420

[34] Yannis Assael, Thea Sommerschield, and Jonathan Prag. 2019. Restoring ancient text using deep learning: A case study on Greek epigraphy. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. 6369–6376.

[35] Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *Advances in Neural Information Processing Systems*. I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30, Curran Associates, Inc. Retrieved from https://proceedings.neurips.cc/paper/2017/file/2d2c8394e31101a261abf1784302bf75-Paper.pdf.

[36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. DOI: http://dx.doi.org/10.48550/ARXIV.1706.03762

[37] Henriette Roued-Cunliffe. 2010. Towards a decision support system for reading ancient documents. *Literary and Linguistic Computing* 25, 4 (10 2010), 365–379. DOI: http://dx.doi.org/10.1093/llc/fqq020 arXiv:https://academic.oup.com/dsh/article-pdf/25/4/365/6192486/fqq020.pdf

[38] Kyeongpil Kang, Kyohoon Jin, Soyoung Yang, Soojin Jang, Jaegul Choo, and Youngbin Kim. 2021. Restoring and mining the records of the Joseon Dynasty via neural language modeling and machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, Online, 4031–4042. DOI: http://dx.doi.org/10.18653/v1/2021.naacl-main.317

[39] Yannis Assael, Thea Sommerschield, Brendan Shillingford, Mahyar Bordbar, John Pavlopoulos, Maria Chatzipanagiotou, Ion Androutsopoulos, Jonathan Prag, and Nando Freitas. 2022. Restoring and attributing ancient texts using deep neural networks. *Nature* 603 (03 2022), 280–283. DOI: http://dx.doi.org/10.1038/s41586-022-04448-z

[40] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*. Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger (Eds.), Vol. 27, Curran Associates, Inc., 3104–3112. Retrieved from https://proceedings.neurips.cc/paper/2014/file/a14ac55a4f27472c5d894ec1c3c743d2-Paper.pdf.

[41] The Packard Humanities Institute. 2005. PHI Greek Inscriptions. Retrieved 19 May 2023 from https://inscriptions.packhum.org/.

[42] Koren Lazar, Benny Saret, Asaf Yehudai, Wayne Horowitz, Nathan Wasserman, and Gabriel Stanovsky. 2021. Filling the gaps in Ancient Akkadian texts: A masked language modelling approach. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Online and Punta Cana, 4682–4691. DOI: http://dx.doi.org/10.18653/v1/2021.emnlp-main.384

[43] Katerina Papavassiliou, Gareth Owens, and Dimitrios Kosmopoulos. 2020. A dataset of Mycenaean Linear B sequences. In *Proceedings of the 12th Language Resources and Evaluation Conference*. European Language Resources Association, Marseille, 2552–2561. Retrieved from https://www.aclweb.org/anthology/2020.lrec-1.311.

[44] Benjamin Snyder, Regina Barzilay, and Kevin Knight. 2010. A statistical model for lost language decipherment. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Uppsala, 1048–1057. Retrieved from https://www.aclweb.org/anthology/P10-1107.

[45] Taylor Berg-Kirkpatrick and Dan Klein. 2011. Simple effective decipherment via combinatorial optimization. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Edinburgh, 313–321. Retrieved from https://www.aclweb.org/anthology/D11-1029.

[46] Jiaming Luo, Yuan Cao, and Regina Barzilay. 2019. Neural decipherment via minimum-cost flow: From Ugaritic to Linear B. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, 3146–3155. DOI: http://dx.doi.org/10.18653/v1/P19-1303

[47] Jiaming Luo, Frederik Hartmann, Enrico Santus, Regina Barzilay, and Yuan Cao. 2021. Deciphering undersegmented ancient scripts using phonetic prior. *Transactions of the Association for Computational Linguistics* 9 (2021), 69–81. DOI: http://dx.doi.org/10.1162/tacl_a_00354

[48] Émilie Pagé-Perron, Maria Sukhareva, Ilya Khait, and Christian Chiarcos. 2017. Machine translation and automated analysis of the Sumerian language. In *Proceedings of the Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*. Association for Computational Linguistics, Vancouver, 10–16. DOI: http://dx.doi.org/10.18653/v1/W17-2202

[49] Ravneet Punia, Niko Schenk, Christian Chiarcos, and Émilie Pagé-Perron. 2020. Towards the first machine translation system for Sumerian transliterations. In *Proceedings of the 28th International Conference on Computational Linguistics*. International Committee on Computational Linguistics, Barcelona, 3454–3460. DOI: http://dx.doi.org/10.18653/v1/2020.coling-main.308

[50] Chanjun Park, Chanhee Lee, Yeongwook Yang, and Heuiseok Lim. 2020. Ancient Korean neural machine translation. *IEEE Access* 8 (2020), 116617–116625. DOI: http://dx.doi.org/10.1109/ACCESS.2020.3004879

[51] M. S. Ruiperez and J. L. Melena. 1996. *The Mycenaean Greeks*. Kardamitsa, Athens.

[52] José L. Melena. 2014. Mycenaean writing. In *A Companion to Linear B: Mycenaean Texts and Their World*. Anna Morpurgo Davies and Yves Duhoux (Eds.), Bibliothèque des Cahiers de l'Institut de Linguistique de Louvain (BCILL), 133, Vol. 3, Peeters, Louvain-la-Neuve, Chapter 17.

[53] Y. Duhoux. 2014. *A Companion to Linear B: Mycenaean Greek Texts and their World*. Bibliothèque des Cahiers de l'Institut de Linguistique de Louvain (BCILL), 133, Vol. 3, Peeters.

[54] F. A. Jorro, F. R. Adrados, and Instituto de Filología (Consejo Superior de Investigaciones Científicas (Spain)). 1985. *DGE*. Consejo Superior de Investigaciones Cientificas, Instituto de Filología. Retrieved from https://books.google.gr/books?id=6ffEt8mdGYkC.

[55] F. A. Jorro, F. R. Adrados, and Instituto de Filología (Consejo Superior de Investigaciones Científicas (Spain)). 1993. *DGE*. Consejo Superior de Investigaciones Cientificas. Retrieved from https://books.google.gr/books?id=64iG3XKSZ2QC.

[56] Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. 2021. A survey of data augmentation approaches for NLP. arXiv:2105.03075v5. Retrieved from https://arxiv.org/abs/2105.03075.

[57] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. 1986. Learning representations by back-propagating errors. *Nature* 323, 6088 (Oct. 1986), 533–536. DOI: http://dx.doi.org/10.1038/323533a0

[58] Felix A. Gers, Nicol N. Schraudolph, and Jürgen Schmidhuber. 2002. Learning precise timing with LSTM recurrent networks. *Journal of Machine Learning Research* 3 (March 2002), 115–143.

[59] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv:1406.1078v3. Retrieved from https://arxiv.org/abs/1406.1078.

[60] Katerina Papavassileiou, Dimitrios Kosmopoulos, and Gareth Owens. 2022. Mycenean linear B sequences - series D. *Zenodo* (2022). DOI: http://dx.doi.org/10.5281/zenodo.7404653

[61] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research)*. Yee Whye Teh and Mike Titterington (Eds.), Vol. 9, PMLR, 249–256. Retrieved from https://proceedings.mlr.press/v9/glorot10a.html.

[62] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In *Proceedings of the 2015 IEEE International Conference on Computer Vision*. DOI: http://dx.doi.org/10.48550/ARXIV.1502.01852

[63] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*. 249–256.

Received 4 March 2022; revised 28 October 2022; accepted 1 November 2022