Embible: Reconstruction of Ancient Hebrew and Aramaic Texts Using Transformers

Niv Fono, Harel Moshayof, Eldar Karol, Itay Asraf, Mark Last

Department of Software and Information Systems Engineering
Ben-Gurion University of the Negev, Beer-Sheva 84105, Israel

{fono,moshayof,eldark,itaias}@post.bgu.ac.il, mlast@bgu.ac.il

Abstract

Hebrew and Aramaic inscriptions serve as an essential source of information on the ancient history of the Near East. Unfortunately, some parts of the inscribed texts become illegible over time. Special experts, called epigraphists, use time-consuming manual procedures to estimate the missing content. This problem can be considered an extended masked language modeling task, where the damaged content can comprise single characters, character n-grams (partial words), single complete words, and multi-word n-grams.

This study is the first attempt to apply the masked language modeling approach to corrupted inscriptions in Hebrew and Aramaic languages, both using the Hebrew alphabet consisting mostly of consonant symbols. In our experiments, we evaluate several transformer-based models, which are fine-tuned on the Biblical texts and tested on three different percentages of randomly masked parts in the testing corpus. For any masking percentage, the highest text completion accuracy is obtained with a novel ensemble of word and character prediction models.

1 Introduction

Every year more and more ancient texts are discovered in both the Hebrew and Aramaic languages throughout the Near East, such as an ancient Hebrew inscription, which was revealed by x-ray measurements on a folded lead tablet in May 2023 (Siegel-Itzkovich, 2023). The analysis of these texts is extremely important for researchers studying the culture and history of the region. As many inscriptions are damaged over time due to earthquakes, fires, political conflicts, and other natural and human-related causes, epigraphists encounter a major challenge in reconstructing the missing parts of these valuable writings. In this non-trivial task, the following difficulties are posed specifically by Hebrew and Aramaic:

1. Language evolution over time. Hebrew and Aramaic are very old languages, both belonging to the group of Semitic languages. The Jewish inhabitants of the Land of Israel have used Classical Hebrew, which is the language of the Bible, from the late eighth to the early sixth centuries BC until they adopted the Aramaic language of the Persian Empire. In the Hellenistic period, around the third century BC, the written Hebrew was revived for various reasons (Schniedewind, 2006). Thus, the inscriptions' period should be taken into account when reconstructing their damaged content.

2. Morphological richness. In contrast to such Indo-European languages as English and French, where conjunctions, articles, and prepositions are separate words, Hebrew and Aramaic use prefixes for the same purpose. For example in Hebrew, the one-letter prefixes Vav, He, and Beth represent the English words 'and', 'the', and 'in', respectively. This makes the tokenization and reconstruction of Hebrew and Aramaic texts significantly more challenging.

Following a study by (Lazar et al., 2021) focusing on Akkadian inscriptions in the cuneiform script (containing hundreds of distinct signs), we define the reconstruction of missing parts in a damaged inscription as a masked language model (MLM) task (Devlin et al., 2019). In this paper, we compare the text completion accuracy of several Transformer-based models including a novel Ensemble approach. The models are trained on two different cases of masked Hebrew text: masked individual characters and masked complete words. The results of extensive evaluation experiments on the variable percentage of randomly masked parts from the Old Testament (Tanakh in Hebrew) indicate the potential usefulness of the proposed

Findings of the Association for Computational Linguistics: EACL 2024, pages 846–852

Ensemble method as a decision-support tool for professional epigraphists specializing in the reconstruction of ancient Hebrew and Aramaic writings. [1]

2 Related Work

There are several studies, which have coped with the problem of restoring damaged writings in various ancient languages. For example, (Fetaya et al., 2020) used RNN models to complete missing tokens in ancient Akkadian texts from the Achaemenid-period Babylonia (539 to 331 BCE). Using the model proposed by the researchers, they reached 85% accuracy in completing the missing token in their test set and 94% accuracy in having the masked token in the top 10 suggestions. In another study related to the Akkadian language (Lazar et al., 2021), the authors use monolingual and multilingual BERT-based models to predict missing signs in Latin transliterations of ancient Mesopotamian documents, originally written on cuneiform clay tablets (2500 BCE - 100 CE). According to their experiments, the probability of a masked token appearing in the top 5 predictions of their model is between 88% and 90%, depending on the document genre. There was also an attempt to reconstruct ancient Greek writings using a bidirectional LSTM aimed at predicting a sequence of missing characters (Assael et al., 2019). This model reached the Character Error Rate (CER) of 30.1%, an improvement of up to 27.2% from suggestions by human experts who were ancient historians.

The above studies suffer from several limitations, which we attempt to overcome in our research. First, they focus on the character prediction subtask rather than on the main epigraphy task of reconstructing the entire multi-word content of a damaged inscription. Consequently, their performance metrics ignore the percentage of accurately completed words, making no distinction between five incorrectly predicted characters in one word and five words with one wrongly predicted character per each word. Moreover, they rarely attempt to combine character prediction and word prediction models and do not study the effect of the masked content amount on the text completion performance. They also ignore an important problem of word separation (whitespace prediction), which exists in many ancient texts but is irrelevant for most masked language models trained on modern documents, where word-based tokenization is straightforward.

To the best of our knowledge, the reconstruction of inscriptions in a consonant-based alphabet, like Hebrew, is not covered by previous studies. Writings mixing two different languages using the same alphabet (e.g., Hebrew and Aramaic) present another unexplored challenge to the text reconstruction task.

The corrupted and omitted text reconstruction problem can also be defined as a string transduction task with monotonic alignments (Ribeiro et al., 2018), which preserves the order of the input (known) characters, without deleting or replacing any of them, and focuses on the insertion of the unknown characters only. Examples of other string transduction tasks include Grammatical Error Correction (GEC) (Rothe et al., 2021), Optical Character OCR post-correction tools (Rijhwani et al., 2020), and Automatic Speech Recognition (ASR) correction approaches (Dutta et al., 2022), with the following important differences from the corrupted text reconstruction problem:

- Correction of some grammatical errors may require deletion and substitution operations, in addition to insertion (Rothe et al., 2021).
- The most common OCR error is confusion between characters of a similar shape (Rijhwani et al., 2020). However, in many corrupted inscriptions, we do not know the shape of missing characters.
- ASR systems may confuse between phonetically similar words (Dutta et al., 2022). Ancient inscriptions, naturally, do not provide any phonetic information.

3 Methodology

In our inscription reconstruction system for Hebrew and Aramaic, we have used the following pre-trained language models:

1. TavBERT (Keren et al., 2022). This BERT-style masked language model is aimed at predicting character sequences rather than contiguous subword tokens, or word-pieces, predicted by most other large language models. The underlying assumption is that individual characters may be more indicative of complex morphological patterns, which are abundant in

[1] Our code is publicly available at https://github.com/harelm4/Embible

Model Name	Num of Epochs	Weight Decay	Batch Size	Learning Rate
TavBERT	20	0	64	5e-5
mBERT	50	0.01	16	2e-6
Distil BERT	50	0.01	32	2e-4
AlephBERT Gimmel	20	0	32	5e-5

Table 1: Language Models

morphologically-rich languages like Hebrew, Arabic, and Turkish. Whitespaces are treated by TavBERT like any other character.

2. mBERT (Devlin et al., 2019). Multilingual BERT (mBERT) is a bi-directional large language model, which is trained simultaneously on texts in 104 languages by masking 15% of subword tokens and then predicting entire masked words only.

3. DistilBERT (Sanh et al., 2019). This is a relatively small language model trained to predict masked tokens (words). To the best of our knowledge, it is one of the few language models that can work with Hebrew texts.

4. AlephBERTGimmel (ABG) (Guetta et al., 2022). This is a language model for modern Hebrew pre-trained on an increased vocabulary size of 128K tokens (word-pieces), which has outperformed the popular HeBERT model (Chriqui and Yahav, 2022) on multiple NLP tasks. The ABG output is a sequence of so-called syntactic words, or morphemes (e.g., some prepositions), which are not necessarily separated by whitespaces in Hebrew and other Semitic languages.

The selected hyperparameter settings of the above models are shown in Table 1. The Number of Epochs for training each model was chosen to minimize the perplexity metric, whereas, in the other settings, we followed the HuggingFace library recommendations. No Aramaic texts were used to pre-train any of these models.

We have evaluated three different configurations of our text completion system for Hebrew inscriptions: Unconstrained Word Completion (UWC), Constrained Word Completion (CWC), and Combined Character and Word Completion (Ensemble).

The UWC approach assumes that we do not know the exact number of masked characters in each damaged fragment of an inscription. If the number of masked whitespaces is also unknown, the number of masked words is assumed to be one. When the number of masked whitespaces is given or predicted, we can deduce the total number of masked words, though the length of each word will still be unknown. To predict the masked word or words, we can apply one of the three word-completion models mentioned above (mBERT, DistilBERT, and ABG). In contrast, the CWC method assumes that we do know the length of each missing word and its boundaries (whitespaces) and, consequently, we can discard any predicted word of incorrect length. CWC can predict a single word of a known length when the whitespaces are not given, and multiple words of a known length otherwise. In addition to insertions, both methods may involve substitutions and deletions of known characters. Due to their simplifying assumptions, we refer to UWC and CWC methods as Baseline 1 and Baseline 2, respectively, and we use them mainly for choosing the most accurate word completion model to be used in the Ensemble method described below.

In addition to the two baselines described above, we introduce a novel method, Ensemble, which represents a more common scenario, where we can reliably estimate the number of masked characters from the inscription font size and geometry, along with the number of masked words and the location of whitespace characters. The Ensemble method combines the character predictions of TavBERT (including whitespaces) and the word predictions of the selected word completion model as follows. First, all masked characters predicted by TavBERT as whitespaces with a probability of 0.50 and higher are treated as known separators between words. Then we use TavBERT to generate the five most probable sequences of missing characters (having the highest average prediction probability). Finally, we search for an overlap between the top predicted character sequences and 1,000 most likely outputs of the selected word prediction model. Word predictions that do not match the known characters in partially masked words or the TavBERT-based word separators are discarded. If the overlap is not empty, we calculate the score of each overlapping prediction as a simple average of the probability scores provided by the two models. Otherwise, we return the top TavBERT predictions with their

originally calculated scores. The Ensemble method involves insertion operations only.

4 Design of Experiments

Our experimental procedure included the following steps:

Step 1 - Data preparation. Since our system is aimed at reconstructing damaged Hebrew inscriptions from the Biblical period, we validated and tested our models on 1,071 verses randomly selected from the Old Testament (*Tanakh* in Hebrew), which was written in Hebrew and Aramaic over several time periods. At least five verses were taken from each Old Testament book. The selected 1,071 verses were split into 535 validation and 536 testing verses. The remaining 22,144 Old Testament verses were used for fine-tuning the pre-trained language models. Diacritical marks (*Nequdot* in Hebrew) and accents (*te'amim* in Hebrew), which were developed and added to the Hebrew Bible only in the Early Middle Ages, were removed from all datasets as irrelevant to inscriptions from the Biblical times.

To explore the effect of the missing content amount on the performance of the fine-tuned models, we created three different versions of the validation and test sets by randomly masking the text in three different percentages: 5%,10%, and 15%. Two different masking strategies were applied. In the first strategy, each word was masked with probability X and if it was not entirely masked, each character in the word was masked with the same probability. In the second strategy, we used the same masking percentages as in the first case, but every word in the text was masked with probability X and also every unmasked character in the text (including white spaces) was masked with probability X.

Step 2 - Model fine-tuning. As described in the methodology section, we performed fine-tuning for the following pre-trained language models: TavBERT, mBERT, DistillBERT, and ABG.

Step 3 - Evaluation. To evaluate our text reconstruction results we use the Hit@K measure:

$Hit@K = (1/N) * \Sigma_{i=1}^{N} 1_{[rank_i <= k]}$

For each predicted element (masked character or word), this metric counts the number of cases where top k predictions include the correct element. In each experiment, we calculate CharHit@K and WordHit@K separately. The option of $k > 1$ indicates that the system can suggest the epigraphists k most likely text completion options along with their estimated probabilities.

5 Evaluation Results

Table 2 in Appendix A evaluates the completion accuracy of three UWC (Baseline 1) models (mBERT, DistillBERT, and ABG), when whitespaces are unknown, and compares them to the Ensemble method. The completion accuracy is measured by the WordHit@1 and WordHit@5 metrics. As expected, there is a slow decline in the performance of each method with an increase in the amount of masked text. However, the Ensemble approach clearly outperforms all Baseline 1 models and its accuracy with 15% Mask is even higher than the accuracy of the best unconstrained model (ABG) with 5% Mask only. Based on the Baseline 1 and 2 results, we have selected ABG as the word prediction model to be used by the Ensemble method alongside TavBERT.

As shown in Table 3 of Appendix A, the accuracy of all methods increases when the whitespaces are known, with Ensemble reaching the WordHit@5 of 0.70 and higher up to the text masking level of 15%. The advantage of the Constrained Word Completion (Baseline 2) models over Baseline 1 models is demonstrated in Tables 4 and 5 of Appendix A for unknown and known whitespaces, respectively. The accuracy of the Ensemble model on our Hebrew corpus is still significantly lower than the accuracy reported in (Lazar et al., 2021) for the Akkadian language. This performance gap can be explained by the differences between the genres of Akkadian texts used in their study and the genre of Biblical verses.

6 Conclusions

It is evident from our experimental results that the proposed ensemble of character and word-based language models is the most beneficial for reconstructing damaged inscriptions in Hebrew and Aramaic. We believe that this approach can be easily extended to writings in morphologically rich and partially deciphered ancient languages like the Ugaritic (Luo et al., 2021). Moreover, the text completion accuracy may be further improved via visual clues from the inscription images. Future research may also include text reconstruction with byte-to-byte language models like ByT5 (Xue et al., 2022) along with a detailed analysis of their reconstruction errors.

7 Limitations

The main limitation of our study is testing the proposed methodology on masked verses from the Old Testament rather than on actual Hebrew and Aramaic inscriptions from the Biblical period. Another limitation is assuming that no information about the possible shape of missing characters is available from the inscription image.

References

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 and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6368–6375, Hong Kong, China. Association for Computational Linguistics.

Avihay Chriqui and Inbal Yahav. 2022. Hebert and hebemo: A hebrew bert model and a tool for polarity analysis and emotion recognition. *INFORMS Journal on Data Science*, 1(1):81–95.

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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Samrat Dutta, Shreyansh Jain, Ayush Maheshwari, Souvik Pal, Ganesh Ramakrishnan, and Preethi Jyothi. 2022. Error correction in asr using sequence-to-sequence models. *arXiv preprint arXiv:2202.01157*.

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):22743–22751.

Eylon Guetta, Avi Shmidman, Shaltiel Shmidman, Cheyn Shmuel Shmidman, Joshua Guedalia, Moshe Koppel, Dan Bareket, Amit Seker, and Reut Tsarfaty. 2022. Large pre-trained models with extra-large vocabularies: A contrastive analysis of hebrew bert models and a new one to outperform them all. *arXiv preprint arXiv:2211.15199*.

Omri Keren, Tal Avinari, Reut Tsarfaty, and Omer Levy. 2022. Breaking character: Are subwords good enough for mrls after all? *arXiv preprint arXiv:2204.04748*.

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*, pages 4682–4691, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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*, 9:69–81.

Joana Ribeiro, Shashi Narayan, Shay Cohen, and Xavier Carreras. 2018. Local string transduction as sequence labeling. In *27th International Conference on Computational Linguistics*, pages 1360–1371. Association for Computational Linguistics (ACL).

Shruti Rijhwani, Antonios Anastasopoulos, and Graham Neubig. 2020. Ocr post correction for endangered language texts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5931–5942.

Sascha Rothe, Jonathan Mallinson, Eric Malmi, Sebastian Krause, and Aliaksei Severyn. 2021. A simple recipe for multilingual grammatical error correction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 702–707.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

William M Schniedewind. 2006. Aramaic, the death of written hebrew, and language shift in the persian period. *Margins of Writing, Origins of Cultures*, pages 137–147.

Judy Siegel-Itzkovich. 2023. Ancient tablet found on mount ebal predates known hebrew inscriptions. *The Jerusalem Post*. Avaialble at: https://www.jpost.com/archaeology/article-743039. 14 May 2023.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306.

A Appendix

WordHit@1	**mask 5%**	**mask 10%**	**mask 15%**
ensemble	0.440	0.317	0.242
ABG	0.147	0.109	0.080
distilbert	0.056	0.042	0.026
mbert	0.045	0.035	0.019
WordHit@5	mask 5%	mask 10%	mask 15%
ensemble	0.503	0.377	0.291
ABG	0.271	0.185	0.148
distilbert	0.108	0.066	0.043
mbert	0.086	0.064	0.040

Table 2: Baseline 1 with Unknown Whitespaces.

WordHit@1	**mask 5%**	**mask 10%**	**mask 15%**
ensemble	0.652	0.623	0.598
ABG	0.251	0.207	0.170
distilbert	0.099	0.078	0.061
mbert	0.086	0.068	0.049
WordHit@5	mask 5%	mask 10%	mask 15%
ensemble	0.739	0.737	0.708
ABG	0.378	0.325	0.285
distilbert	0.146	0.124	0.102
mbert	0.139	0.111	0.094

Table 3: Baseline 1 with Known Whitespaces.

WordHit@1	mask 5%	mask 10%	mask 15%
ensemble	0.440	0.317	0.242
ABG	0.188	0.128	0.099
distilbert	0.072	0.048	0.034
mbert	0.059	0.043	0.029
WordHit@5	mask 5%	mask 10%	mask 15%
ensemble	0.503	0.377	0.291
ABG	0.271	0.185	0.148
distilbert	0.107	0.075	0.148
mbert	0.093	0.073	0.052
CharHit@1	mask 5%	mask 10%	mask 15%
ensemble	0.589	0.372	0.293
ABG	0.367	0.215	0.175
distilbert	0.181	0.092	0.083
mbert	0.155	0.090	0.078
CharHit@5	mask 5%	mask 10%	mask 15%
ensemble	0.696	0.452	0.365
ABG	0.556	0.368	0.315
distilbert	0.369	0.224	0.189
mbert	0.342	0.215	0.188

Table 4: Baseline 2 with Unknown Whitespaces.

WordHit@1	**mask 5%**	**mask 10%**	**mask 15%**
ensemble	0.712	0.616	0.600
ABG	0.337	0.295	0.253
distilbert	0.127	0.116	0.099
mbert	0.128	0.103	0.089
WordHit@5	mask 5%	mask 10%	mask 15%
ensemble	0.779	0.728	0.710
ABG	0.475	0.429	0.396
distilbert	0.190	0.167	0.159
mbert	0.182	0.160	0.150
CharHit@1	mask 5%	mask 10%	mask 15%
ensemble	0.692	"	0.577"
ABG	0.578	0.421	0.367
distilbert	0.271	0.194	0.168
mbert	0.261	0.191	0.164
CharHit@5	mask 5%	mask 10%	mask 15%
ensemble	0.909	0.691	0.633
ABG	0.870	0.665	0.617
distilbert	0.512	0.380	0.343
mbert	0.497	0.355	0.342

Table 5: Baseline 2 with Known Whitespaces.