

## ***Philo-1-preview***

# **Un modello T5-Base per l'emendazione dei testi antichi**

Giuseppe Ferrara<sup>1</sup>

<sup>1</sup> Università degli Studi di Siena, Italia - giuseppe.ferrara@student.unisi.it

### **ABSTRACT (ITALIANO)**

In questo paper, si presenta *Philo-1-preview*, un *Large Language Model* di tipo *Seq2Seq* di 297 milioni di parametri ottenuto mediante un'operazione di *fine-tuning* condotta su *PhilTa*, una variante *encoder-decoder* multilingue di T5-Base pre-addestrata sul greco antico e sul latino, e specializzato nell'individuazione e nella correzione degli errori di maiuscola, minuscola, pronuncia, dittografia e aplografia, combinati o isolati, che si possono riscontrare nel testo tradito di molte opere antiche. Il suo sviluppo è stato guidato da un approccio al problema diverso da quelli già sperimentati in progetti precedenti, quali *Bentley* e *Ianus*, di cui *Philo-1-preview* e il suo sistema di *front-end Ianus AI* rappresentano i diretti discendenti, e *Latin BERT* e *Logion*: l'emendazione non viene più trattata come un processo meccanico da risolvere con un algoritmo *rule-based* o come un *task* di *fill-mask*, ma come un *task* di generazione di testo (*text-to-text*) combinato con un'operazione di *denoising*. I *dataset* utilizzati per il *fine-tuning* del modello sono stati ricavati dai *corpora* di testi greci e latini presenti su *Perseus Digital Library*, dopo alcune opportune operazioni di *pre-processing* volte a isolarne le parti effettivamente scritte nelle lingue di interesse e a normalizzarne i criteri di trascrizione. In fase di valutazione, *Philo-1-preview* ha mostrato prestazioni molto elevate nel compito per cui è stato progettato, con un *BLEU score* di 67.50 e un *cross-entropy loss* di 0.078. Il confronto con *Latin BERT* e *Logion* ha evidenziato che *Philo-1-preview* fornisce correzioni dei testi antichi di qualità superiore a quelle degli altri modelli, almeno per gli errori su cui è stato condotto il *fine-tuning*. Per superare i limiti di questo modello, si è già iniziato a lavorare allo sviluppo di un modello BART che sia in grado di identificare e correggere, in autonomia, tutte le tipologie di errori riscontrabili nella tradizione delle opere antiche.

**Parole chiave:** filologia digitale; Large Language Model; Ianus AI; Philo-1-preview; emendazione

### **ABSTRACT (ENGLISH)**

*Philo-1-preview. A T5-Base model for the emendation of ancient texts*

In this paper, I present *Philo-1-preview*, a *Seq2Seq Large Language Model* with 297 million parameters obtained through a fine-tuning process conducted on *PhilTa*, a multilingual encoder-decoder variant of T5-Base pre-trained on Ancient Greek and Latin. It is specialized in identifying and correcting majuscule and minuscule scripts, pronunciation, dittography and haplography errors, whether combined or isolated, which can be found in the textual transmission of many ancient works. Its development has been guided by an approach to the problem different from those already tested in previous projects, such as *Bentley* and *Ianus*, of which *Philo-1-preview* and its front-end system *Ianus AI* are the direct descendants, and *Latin BERT* and *Logion*. In this new approach, emendation is no longer treated as a mechanical process to be solved with a rule-based algorithm or as a fill-mask task, but rather as a text-generation (*text-to-text*) task combined with a denoising operation. The datasets used for the model's fine-tuning were derived from the Greek and Latin text corpora in the *Perseus Digital Library*, after some pre-processing steps aimed at isolating the parts actually written in the languages of interest and to normalize their transcription criteria. During the evaluation phase, *Philo-1-preview* showed very high performance in the task for which it was designed, with a BLEU score of 67.50 and a cross-entropy loss of 0.078. A comparison with *Latin BERT* and *Logion* revealed that *Philo-1-preview* provides higher-quality corrections of ancient texts than the other models, at least for the types of errors on which it was fine-tuned. In order to overcome the limitations of this model, work has already begun on the development of a BART model capable of autonomously identifying and correcting all the types of errors that can be found in the textual tradition of ancient works.

**Keywords:** digital philology; Large Language Model; Ianus AI; Philo-1-preview; emendation

## **1. INTRODUZIONE**

In questo contributo, si presenta *Philo-1-preview* la versione beta di un *Large Language Model* (LLM) di tipo *Seq2Seq* basato sull'architettura T5 (*Text-to-Text Transfer Transformer*) (Raffel et al., 2019) pensato

per semplificare l'emendazione dei testi antichi, greci e latini.<sup>1</sup> Lo sviluppo di tale modello si è reso necessario dopo aver constatato la scarsa diffusione di tecnologie che siano in grado di assistere, con profitto, i filologi classici in questa fase del processo di edizione di un'opera antica. I LLMs sono stati finora impiegati principalmente per agevolare processi di analisi testuale caratteristici del *Natural Language Processing* (NLP), come il *PoS Tagging*, il *Dependency Parsing* e la lemmatizzazione, mentre il loro impiego per la ricostruzione degli originali dei testi antichi è limitato a pochi casi isolati, alcuni dei quali esulano dall'ambito delle lingue classiche:

1. *MsBERT*, un modello BERT pre-addestrato su un *dataset* comprensivo di trascrizioni di manoscritti ebraici, del *Pre-Print Rabbinic Corpus* e del *Comprehensive Rabbinic Corpus*, per l'integrazione dei testi ebraici trasmessi in forma frammentaria (Shmidman et al., 2024);
2. una RNN addestrata sui *Coptic SCRIPTORIUM Corpora* per l'integrazione delle lacune dei manoscritti coptici (Levine et al., 2024);
3. *Pythia* (Assael et al., 2019) e *Ithaca* (Assael et al., 2022), due modelli, rispettivamente un LSTM di tipo *Seq2Seq* e un modello basato sull'architettura *Transformer*, per l'integrazione delle iscrizioni greche pervenute mutili;
4. *Latin BERT* (Bamman & Burns, 2020), un modello BERT pre-addestrato su vari *corpora* di testi latini (*Corpus Thomisticum*, *Internet Archive*, *The Latin Library*, *Patrologia Latina*, *Perseus Digital Library*, *Vicipaedia*) per l'integrazione delle lacune in essi presenti;
5. *Logion* (Cowen-Breen et al., 2023; Graziosi et al., 2023), un modello BERT pre-addestrato su una versione ampliata del *corpus* di testi greci premoderni di Singh et al. (2021) per l'emendazione dei testi greci.

Tra questi modelli, quelli specializzati nei domini del greco antico e del latino sono caratterizzati da livelli di *top-1 accuracy* non ancora soddisfacenti nei *task* di emendazione o integrazione: 32.6% per *Pythia* (Assael et al., 2022: 282); 61.8% per *Ithaca* (Assael et al., 2022: 282); 33.1% per *Latin BERT* (Bamman & Burns, 2020: 4–6); 58.3% per *Logion* (Cowen-Breen et al., 2023: 8).

*Philo-1-preview* si inserisce nel panorama dei modelli attualmente disponibili apportando alcune novità significative agli approcci già sperimentati. L'impostazione adottata per sviluppare il modello affronta il problema della ricostruzione del testo originario di un'opera antica come un *task* di generazione di testo (*text-to-text*) combinato con un'operazione di cancellazione del rumore presente nell'*input* del modello (*denoising*). *Latin BERT* e *Logion* trattano il medesimo problema come un *task* di *fill-mask* che rispecchia il *task* di *Masked Language Modeling* (MLM) usato durante il loro pre-addestramento (Bamman & Burns, 2020: 4–6; Cowen-Breen et al., 2023: 7–8): in altri termini, essi cercano di predire la parola più adatta a colmare una lacuna artificialmente introdotta, all'interno della frase, in corrispondenza di una parola considerata errata. Da un lato, tale approccio permette di utilizzare direttamente i modelli pre-addestrati per l'emendazione dei testi antichi senza necessariamente fine-tunarli, dal momento che il compito per cui vengono impiegati è coerente con la metodologia utilizzata durante il *pre-training* per far acquisire al modello le competenze linguistiche desiderate. Dall'altro, un simile approccio richiede uno sforzo implementativo aggiuntivo, rendendo indispensabile lo sviluppo di un sistema che permetta al modello di individuare correttamente le parole errate presenti in una data frase, nel caso in cui non le si voglia segnalare esplicitamente tramite il *token* [MASK]. Trattare, invece, l'emendazione di un testo antico come un *task* di *denoising* attraverso un modello *Seq2Seq* basato sull'architettura T5 permette di implementare l'individuazione automatica, da parte del modello, degli errori presenti in quel testo.

Lo sviluppo di *Philo-1-preview* rappresenta il risultato di un lavoro di ricerca più ampio articolato in tre tappe:

1. l'implementazione, nel 2020, di *Bentley*, un *software rule-based* per la correzione degli errori di minuscola presenti nei testi latini (Ferrara, 2023: 85);
2. lo sviluppo, nel 2023, di *Ianus*, una nuova versione di *Bentley* che estende la correzione degli errori di minuscola anche al greco antico e introduce la correzione degli errori di maiuscola e pronuncia in entrambe le lingue (Ferrara, 2023);
3. l'implementazione, iniziata nel 2024, di *Ianus AI*, un *software* per un'emendazione dei testi antichi assistita dall'Intelligenza Artificiale.

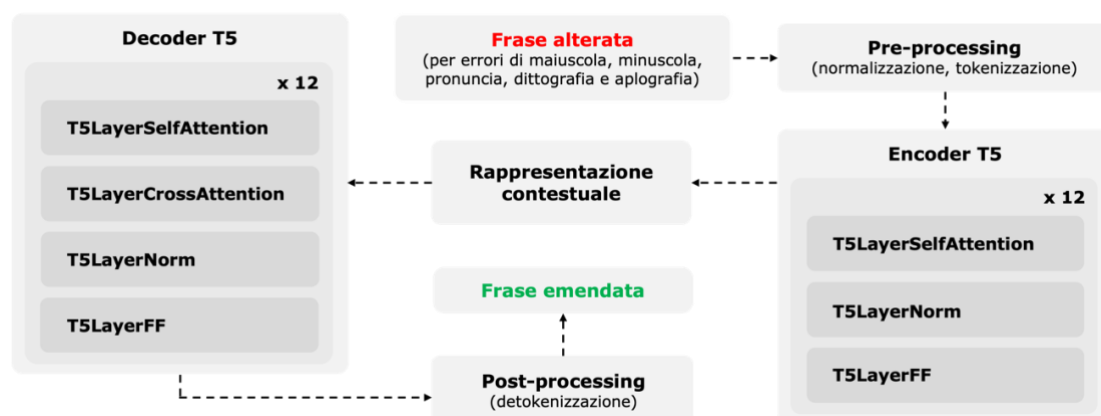
---

<sup>1</sup> Una volta terminata la fase di *beta testing*, il modello e il suo sistema di *front-end Ianus AI* verranno distribuiti sotto licenza GNU GPLv3 rispettivamente su <https://huggingface.co/giuseppeferrara/Philo-1-preview> (cons. 10/04/2025) e <https://github.com/giuseppeferrara/ianus-ai.git> (cons. 10/04/2025).

L'evoluzione metodologica segnata dall'ultima tappa di questo percorso, che ha sostituito gli algoritmi deterministici implementati nelle due versioni precedenti del *software* con i LLMs, si è resa indispensabile per migliorare la capacità predittiva di *Ianus* attraverso la valutazione del contesto della frase per l'elaborazione delle congetture e per rendere la nuova versione del *software* capace di individuare autonomamente le corruzioni presenti nel testo tradito.

## 2. ARCHITETTURA DEL MODELLO, DATASET E FINE-TUNING

*Philo-1-preview* è un modello di 297 milioni di parametri ottenuto attraverso un'operazione di *fine-tuning* condotta su *PhilTa* (Riemenschneider & Frank, 2023), una variante *encoder-decoder* multilingue del modello T5-Base (Raffel et al., 2019) pre-addestrata su vari *corpora* di testi greci e latini (*Open Greek & Latin Project*, *Greek Medieval Texts* di CLARIN, *Patrologia Graeca*, *Corpus Corporum*, *Project Gutenberg*). Il *fine-tuning* ha avuto come obiettivo dotare il modello della capacità di individuare e correggere autonomamente gli errori presenti nel testo tradito delle opere antiche.



**Figura 1. Architettura e flusso logico di *Philo-1-preview* e di *Ianus AI*, il suo sistema di *front-end*.**

L'architettura di *Philo-1-preview*, integrata con il flusso logico del modello e del suo sistema di *front-end*, è riportata in Fig.1. La frase alterata per la presenza di errori di maiuscola, minuscola, pronuncia, dittografia e aplografia (di singole lettere) viene pre-processata ed entra nell'*encoder* del modello, che ne produce una rappresentazione contestuale tenendo conto delle relazioni tra i singoli *token*. Questa rappresentazione viene inviata al *decoder* del modello che predice la sequenza di *token* della frase emendata. Quest'ultima verrà convertita in una stringa testuale grazie alla detokenizzazione che ha luogo durante il *post-processing*.

I *dataset* utilizzati nelle fasi di *training*, validazione e valutazione sono stati ricavati dai *corpora* di testi greci e latini presenti su *Perseus Digital Library*, dopo alcune opportune operazioni di *pre-processing*:

1. eliminazione degli elementi paratestuali in essi presenti (tag XML, commenti editoriali, note, apparati critici);
2. normalizzazione dei testi, articolata in:
  - a) sostituzione di tutte le lettere maiuscole con le corrispondenti minuscole;
  - b) eliminazione delle lettere ramiste da quelli latini;
  - c) sostituzione della iota sottoscritta con quella ascritta in quelli greci;
  - d) rimozione dei segni diacritici nelle lettere interessate da possibili fraintendimenti da quelli greci.

In questo modo, si sono ottenuti due file CSV, uno per il greco antico e uno per il latino, contenenti, come *record*, coppie di frasi, di cui una alterata (per errori di maiuscola, minuscola, pronuncia, dittografia e aplografia isolati o combinati) e l'altra corretta, di lunghezza massima pari a 15 parole. Le regole per produrre le frasi nella loro versione alterata sono state, in parte, ricavate dall'algoritmo di *Ianus* e, in parte, formulate *ex novo*. Il *dataset* complessivo (con 1028111 *record*) è stato suddiviso in *dataset* di *training* (90% dei *record* totali) e *dataset* di valutazione (10% dei *record* totali) (Abraham et al., 2024: 390).

Gli iperparametri da utilizzare durante il *fine-tuning*, riportati in Tab.1, sono stati trovati attraverso un'operazione di ottimizzazione condotta con *Optuna* e articolata in 10 *trial* effettuati su un *subset* pari al 10% del *dataset* finale. La loro ricerca e il *training* effettivo del modello, della durata di circa 13 ore, sono stati svolti su una GPU NVIDIA A100.

Iperparametro	Valore
Numero di epoche	5
Learning rate	0.0002123818291492695
Batch size	16
Weight decay	0.2984303550950661

**Tabella 1. Iperparametri utilizzati durante il processo di *fine-tuning* di *PhilTa*.**

### 3. VALUTAZIONE ED EFFICACIA DEL MODELLO

Le *performance* di *Philo-1-preview* nel *task* di interesse sono state valutate attraverso due metriche: il *BLEU* (*Bilingual Evaluation Understudy*) *score* (Papineni et al., 2001), utilizzato per ottenere una misura quantitativa della qualità delle correzioni proposte dal modello mediante il confronto tra la frase prodotta in *output* e la frase corretta di riferimento; il *cross-entropy loss*, che, confrontando la distribuzione di probabilità prevista dal modello con quella reale dei *token* della frase corretta, fornisce una misura degli scarti del modello stesso (Rao & McMahan, 2019: 46–48). Per massimizzare l’attendibilità della prima delle due metriche, si sono ignorati i *token* di *padding*.

Metrica	<i>PhilTa</i>	<i>Philo-1-preview</i>	Variazione di <i>performance</i>
<i>BLEU score</i>	2.03	67.50	+65.47
<i>Cross-entropy loss</i>	3.25	0.078	-3.17

**Tabella 2. Confronto tra i valori delle metriche di valutazione delle *performance* di *PhilTa* e *Philo-1-preview*.**

Più in dettaglio, la valutazione di *Philo-1-preview* si è articolata in due fasi. La prima ha avuto come obiettivo la verifica del corretto aggiornamento dei pesi del modello prodotto dal processo di *fine-tuning*. A tale scopo, si è effettuato un confronto tra le *performance* di *PhilTa* e quelle di *Philo-1-preview* sul medesimo *dataset* di valutazione (i risultati sono riportati in Tab.2). *Philo-1-preview* ha ottenuto un *BLEU score* di 67.50, contro il 2.03 di *PhilTa*, dimostrando di aver effettivamente acquisito, attraverso il processo di *fine-tuning*, la capacità di produrre correzioni del testo analizzato di qualità molto elevata. Il *cross-entropy loss* del modello è stato di 0.078, contro il 3.25 di *PhilTa*.

La seconda fase ha avuto come obiettivo la valutazione dell’effettivo contributo di *Philo-1-preview* al processo di emendazione dei testi antichi, attraverso il suo confronto con i modelli *gold-standard* sopra menzionati: *Latin BERT* e *Logion*. Si ricordi, per inciso, che questi ultimi, al contrario di *Philo-1-preview*, non sono stati *fine-tunati*, ma soltanto *pre-addestrati*. Questo, tuttavia, non invalida il confronto e i risultati ottenuti per due ragioni: in primo luogo, perché gli sviluppatori di *Latin BERT* e *Logion* presentano esplicitamente questi modelli come destinati, anche o esclusivamente, all’emendazione dei testi antichi nella loro versione *pre-addestrata*; in secondo luogo, perché la valutazione, nel loro caso, è stata effettuata non sul *task* di generazione di testo e di *denoising*, come per *Philo-1-preview*, ma sul *task* di *fill-mask* per cui essi stessi sono stati ottimizzati. A questo scopo, si è resa necessaria una modifica dei *record* del *dataset* di valutazione per poterne riadattare la struttura al compito di *gap-filling* e poterli utilizzare come *input* di *Latin BERT* e *Logion*: le frasi alterate contenute nelle coppie di frasi selezionate (per il resto rimaste inalterate) sono state modificate sostituendo le loro parole errate con il *token* speciale [MASK].

Modello	<i>BLEU score</i>
<i>Philo-1-preview</i>	67.50
<i>Latin BERT</i>	0.76
<i>Logion</i>	0.81

**Tabella 3. Confronto tra i *BLEU score* di *Philo-1-preview*, *Latin BERT* e *Logion*.**

I risultati del confronto, riportati in Tab.3, mostrano che *Philo-1-preview* è, tra i modelli disponibili, quello che fornisce correzioni dei testi antichi più qualitative, almeno per gli errori di maiuscola, minuscola, pronuncia, dittografia e aplografia (di singole lettere). Nella correzione di queste categorie di errori, intesa come predizione delle parole più adatte a colmare le lacune introdotte al posto delle parole errate, *Latin BERT* e *Logion* hanno ottenuto un *BLEU score* molto basso, rispettivamente di 0.76 e 0.81.

#### 4. IL SISTEMA DI *FRONT-END* E LA SUA GUI

The figure displays two versions of the Ianus AI web interface. The left version shows the initial state where the user input field is empty. The right version shows the interface after a correction, with the corrected text displayed in a green box below the input area. Both versions feature a header with the Ianus AI logo and 'Beta 1' label, followed by a description of the service. Below this are three dropdown menus for selecting the model (Philo-1-preview), language (Latino), and correction mode (Correzione generale). A 'Parametri di generazione' dropdown is also present. The input area is labeled 'Inserisci il testo latino da emendare:'. The correction button is a red 'Correggi' button.

**Figura 2.** Interfaccia grafica di *Ianus AI*, il sistema di *front-end* di *Philo-1-preview*.

Per rendere il modello facilmente interrogabile, è stato realizzato anche un sistema di *front-end* dotato di una *Graphic User Interface* (GUI) semplice e intuitiva (vd. Fig.2), utilizzando il *framework open-source Streamlit*. I principi che hanno guidato lo sviluppo della GUI sono stati:

1. creare una gerarchia visiva tra gli elementi dell'interfaccia;
2. implementare un design pulito e minimalista;
3. ridurre la curva di apprendimento dell'utente per massimizzare l'usabilità dell'applicazione;
4. responsività.

Il primo principio si è tradotto nell'utilizzo, per le varie componenti dell'interfaccia, di dimensioni tipografiche diverse, finalizzate a rendere facilmente distinguibili gli elementi descrittivi da quelli funzionali. Il secondo principio si è tradotto nell'utilizzo di un *layout* semplice e arioso attraverso una corretta gestione degli spazi bianchi, per far focalizzare l'attenzione dell'utente sui *task* principali dell'applicazione. Il terzo principio è stato implementato attraverso un sistema di *feedback* offerti all'utente sotto forma di avvisi e messaggi di errore e attraverso un'interfaccia lineare in grado di guidare l'utente durante l'intero processo di correzione. Il quarto principio si è tradotto nello sviluppo di un'interfaccia dotata di un *layout* fluido e flessibile, in grado di adattarsi, in autonomia, alle differenti risoluzioni degli schermi dei dispositivi, senza sacrificare la funzionalità dell'applicazione.

L'utente, dopo aver selezionato il modello da utilizzare per l'emendazione, la lingua del testo immesso e la modalità di correzione (generale, che corregge autonomamente l'intero testo immesso, o mirata, che corregge soltanto le parole del testo segnalate al modello dall'utente), inserisce l'*input* testuale corrotto in un *widget TextArea* e, cliccando sul pulsante *Correggi*, innesca le operazioni di *pre-processing*, inferenza e *post-processing*, che porteranno all'elaborazione del testo emendato. Quest'ultimo verrà mostrato in un riquadro apposito che comparirà, dinamicamente, sotto il *widget TextArea*.

#### 5. CONCLUSIONI: LIMITI E SVILUPPI FUTURI

Per concludere la presentazione di questa ricerca, è opportuno soffermarsi brevemente sui suoi limiti, in quanto essi possono essere considerati il punto di partenza per ulteriori sviluppi futuri.

Il primo limite è rappresentato dal fatto che *Philo-1-preview* è stato ottimizzato soltanto per la correzione di alcune categorie di errori filologici. Per espanderne il raggio di azione, si sta già lavorando allo sviluppo di un modello più ampio, basato su BART (*Bidirectional and Auto-Regressive Transformers*) (Lewis et al., 2019), che sia in grado di identificare e correggere tutte le tipologie di errori riscontrabili nella tradizione di

un'opera antica. La scelta di questo modello dipende dalla sua particolare architettura, che combina la capacità di comprensione profonda del testo, tipica di modelli bidirezionali come BERT (Devlin et al., 2019), e l'abilità di generazione sequenziale, caratteristica di modelli autoregressivi come GPT (*Generative Pre-trained Transformer*) (Radford & Narasimhan, 2018; Lewis et al., 2019: 1–2).

Il secondo limite è rappresentato dalle dimensioni del modello in termini di numero di parametri. Per quanto *Philo-1-preview* non sia, tecnicamente, tra i modelli più piccoli, aumentarne il numero complessivo di parametri potrebbe permettergli di raggiungere una comprensione più profonda delle relazioni sintattiche e delle sfumature semantiche presenti nei testi greci e latini e di produrre correzioni di qualità ancora superiore (Kaplan et al., 2020; Gazit, 2024: 174). Questo obiettivo potrà essere raggiunto apportando alcune modifiche all'architettura del modello, al processo di tokenizzazione e alla fase di *training*.

Il terzo limite risiede nel fatto che non si è sviluppato *ex novo* un modello specifico per il greco antico e per il latino, ma si è utilizzato un modello pre-addestrato, del quale non è stato possibile verificare, in via preliminare, l'effettiva qualità dell'intero processo di *pre-training*, fondamentale per garantire la capacità del modello di produrre inferenze corrette. Questa criticità verrà superata attraverso lo sviluppo, controllato in ogni sua fase, del modello BART cui si è fatto cenno sopra.

Il quarto limite della ricerca risiede nell'aver utilizzato, per il *fine-tuning* di *PhilTa*, un *corpus* ristretto di testi greci e latini, che non comprende tutti quelli effettivamente disponibili *online* sotto licenza *open-source*. Questa scelta, da un lato, potrebbe limitare la capacità dell'attuale versione del modello di fornire correzioni di testi di epoche, stili, generi letterari (e dialetti, nel caso del greco antico) sottorappresentati e di termini rari o tecnici poco frequenti o assenti nel *corpus* di *Perseus Digital Library*, dall'altro è stata dettata dalla necessità di validare l'approccio adottato nello sviluppo di *Philo-1-preview*, prima di estenderlo a *dataset* più ampi. Si tenga comunque conto del fatto che il numero totale di *record* ottenuti si è rivelato sufficiente per sviluppare un modello dotato di prestazioni elevate nel *task* di interesse.

## BIBLIOGRAFIA

- Abraham, A., Bajaj, A., & Hanne, T. (A c. Di). (2024). Machine learning solutions. International Conference on Intelligent Systems Design and Applications, Cham. Springer.
- Assael, Y., Sommerschild, T., & Prag, J. (2019). Restoring ancient text using deep learning: A case study on Greek epigraphy (arXiv:1910.06262). arXiv. <http://arxiv.org/abs/1910.06262>.
- Assael, Y., Sommerschild, T., Shillingford, B., Bordbar, M., Pavlopoulos, J., Chatzipanagiotou, M., Androutsopoulos, I., Prag, J., & de Freitas, N. (2022). Restoring and attributing ancient texts using deep neural networks. *Nature*, 603(7900), 280–283. <https://doi.org/10.1038/s41586-022-04448-z>.
- Bamman, D., & Burns, P. J. (2020). Latin BERT: A Contextual Language Model for Classical Philology (arXiv:2009.10053). arXiv. <http://arxiv.org/abs/2009.10053>.
- Cowen-Breen, C., Brooks, C., Haubold, J., & Graziosi, B. (2023). Logion: Machine Learning for Greek Philology (arXiv:2305.01099). arXiv. <https://doi.org/10.48550/arXiv.2305.01099>.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (arXiv:1810.04805). arXiv. <http://arxiv.org/abs/1810.04805>.
- Ferrara, G. (2023). Ianus 2.0. Corruttele filologiche, congetture digitali. La memoria digitale: forme del testo e organizzazione della conoscenza. Atti del XII Convegno Annuale AIUCD, 85–91. [https://amsacta.unibo.it/id/eprint/7721/1/atti\\_aiucd\\_2023.pdf](https://amsacta.unibo.it/id/eprint/7721/1/atti_aiucd_2023.pdf).
- Gazit, L. (con un contributo di Ghaffari, M., & Saxena, A.). (2024). Mastering NLP from Foundations to LLMs: Apply advanced rule-based techniques to LLMs and solve real-world business problems using Python (1a ed.). Packt Publishing Limited.
- Graziosi, B., Haubold, J., Cowen-Breen, C., & Brooks, C. (2023). Machine Learning and the Future of Philology: A Case Study. *TAPA*, 153(1), 253–284. <https://doi.org/10.1353/apa.2023.a901022>.
- Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., & Amodei, D. (2020). Scaling Laws for Neural Language Models (arXiv:2001.08361). arXiv. <https://doi.org/10.48550/arXiv.2001.08361>.
- Levine, L., Li, C., BremerMcCollum, L., Wagner, N., & Zeldes, A. (2024). Lacuna Language Learning: Leveraging RNNs for Ranked Text Completion in Digitized Coptic Manuscripts. *Proceedings of the 1st Workshop on Machine Learning for Ancient Languages (ML4AL 2024)*, 61–70. <https://doi.org/10.18653/v1/2024.ml4al-1.8>.

- Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2019). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension (arXiv:1910.13461). arXiv. <https://doi.org/10.48550/arXiv.1910.13461>.
- Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2001). BLEU: A method for automatic evaluation of machine translation. Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL '02, 311. <https://doi.org/10.3115/1073083.1073135>.
- Radford, A., & Narasimhan, K. (2018). Improving Language Understanding by Generative Pre-Training. <https://api.semanticscholar.org/CorpusID:49313245>.
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2019). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer (Version 4). arXiv. <https://doi.org/10.48550/ARXIV.1910.10683>.
- Rao, D., & McMahan, B. (2019). Natural language processing with PyTorch: Build intelligent language applications using deep learning (First edition). O'Reilly.
- Riemenschneider, F., & Frank, A. (2023). Exploring Large Language Models for Classical Philology (arXiv:2305.13698). arXiv. <http://arxiv.org/abs/2305.13698>.
- Shmidman, A., Shmidman, O., Gershuni, H., & Koppel, M. (2024). MsBERT: A New Model for the Reconstruction of Lacunae in Hebrew Manuscripts. Proceedings of the 1st Workshop on Machine Learning for Ancient Languages (ML4AL 2024), 13–18. <https://doi.org/10.18653/v1/2024.ml4al-1.2>.
- Singh, P., Rutten, G., & Lefever, E. (2021). A Pilot Study for BERT Language Modelling and Morphological Analysis for Ancient and Medieval Greek. Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature, 128–137. <https://doi.org/10.18653/v1/2021.latechclfl-1.15>.