BIOREADER: a Retrieval-Enhanced Text-to-Text Transformer for Biomedical Literature

Giacomo Frisoni, Miki Mizutani, Gianluca Moro, Lorenzo Valgimigli
Department of Computer Science and Engineering (DISI)
University of Bologna, Via dell'Università 50, I-47522 Cesena, Italy
{giacomo.frisoni, gianluca.moro, lorenzo.valgimigli}@unibo.it
{miki.mizutani}@studio.unibo.it

Abstract

The latest batch of research has equipped language models with the ability to attend over relevant and factual information from non-parametric external sources, drawing a complementary path to architectural scaling. Besides mastering language, exploiting and contextualizing the latent world knowledge is crucial in complex domains like biomedicine. However, most works in the field rely on general-purpose models supported by databases like Wikipedia and Books. We introduce BIOREADER[1], the first retrieval-enhanced text-to-text model for biomedical natural language processing. Our domain-specific T5-based solution augments the input prompt by fetching and assembling relevant scientific literature chunks from a neural database with ≈60 million tokens centered on PubMed. We fine-tune and evaluate BIOREADER on a broad array of downstream tasks, significantly outperforming several state-of-the-art methods despite using up to 3x fewer parameters. In tandem with extensive ablation studies, we show that domain knowledge can be easily altered or supplemented to make the model generate correct predictions bypassing the retraining step and thus addressing the literature overload issue.

1 Introduction

In the last decade, deep learning advancements have boosted the development of many solutions for effectively extracting knowledge from biological data (Domeniconi et al., 2014a, 2016a) and biomedical literature (di Lena et al., 2015)—widely accessible through repositories such as PubMed, PMC, and ScienceDirect. Large pre-trained language models (PLMs) have become the dominant NLP paradigm, achieving unprecedented results in a panoply of tasks, from named entity recognition (Lee et al., 2020) and semantic parsing (Frisoni et al., 2021, 2022b) to information retrieval (Moro and Valgimigli, 2021) and document summarization (Moro et al., 2022).

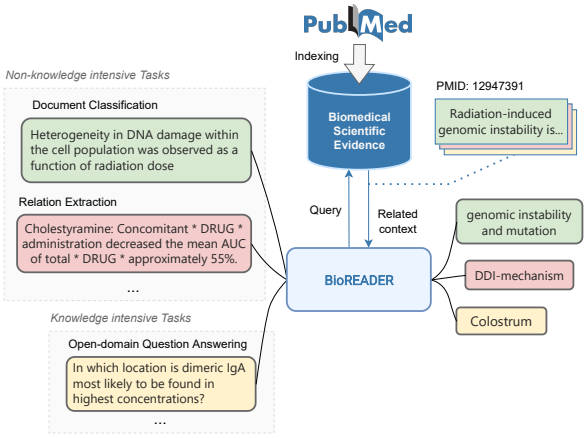

Figure 1: Illustration of BIOREADER, a text-to-text model conditioned on scientific evidence retrieved from an explicit PubMed-based datastore. Every biomedical task is cast as translating text spans with the help of external domain knowledge retrieved on the fly.

To justify this success, PLMs have been shown to implicitly hold a substantial amount of in-depth knowledge in their parameters (Petroni et al., 2019; Davison et al., 2019), resulting from self-supervised learning on extreme-scale text corpora.

Efforts to this point have mainly focused on predictably improving NLP performance by increasing datasets, training compute, or model sizes. Notably, the most recent Transformer-based solutions reach up to 10^{11} parameters (Brown et al., 2020; Rae et al., 2021), with benefits due to extended memorization of training data (Carlini et al., 2021; Tirumala et al., 2022). However, encoding all factual and domain-specific competencies into opaque weight matrices is inefficient, especially for specialized, dynamic, and trust-demanding fields like biomedicine (Lötsch et al., 2021)—where the volume of scientific publications evolves and grows continuously (Landhuis, 2016). Indeed, capturing more world facts requires training ever-

[1] https://github.com/disi-unibo-nlp

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 5770 - 5793
December 7-11, 2022

larger networks, which can be prohibitively slow or expensive. Similarly, changing what a PLM knows entails retraining the entire model with new documents. Moreover, high-dimensional non-interpretable parametric spaces make it difficult to determine what knowledge is stored where, to update the theoretical background, or to provide provenance for decisions.

Recent developments in the field (Lewis et al., 2020b; Nakano et al., 2021; Borgeaud et al., 2021) have reversed the architectural scaling trend by showing that smaller PLMs can perform on par with massive models if we augment them with a way to search for external information. The key intuition is following a *retrieve-then-predict* approach by asking the PLM to directly fetch potentially relevant unlabeled world knowledge—even structured (Yasunaga et al., 2021)—from highly-comprehensive datastores, and use it as additional context during inference. Remarkably, semi-parametric (or hybrid) contributions combine "closed-book" (parametric-only) and "open-book" (retrieval-based) methods to complement each source. They allow for revising or even supplementing knowledge dynamically, treating the latter in a more modular and interpretable way. On the other hand, semi-parametric models have so far been only investigated for general-domain knowledge bases and NLP tasks, e.g., context-free question answering conditioned on Wikipedia and Books evidence. Based on previous publications (Beltagy et al., 2019; Lee et al., 2020; Gu et al., 2022), the word distribution shift from general corpora to health informatics corpora prevents or seriously limits the direct application of such models to biomedical NLP.

In this paper, we introduce BIOREADER, the first *retrieval-enhanced transformer* for biomedical literature, empowered by a differentiable access towards a large-scale text memory grounded on PubMed (≈60M tokens). We continue and build on a broad spectrum of retrieval work in the research community (Li et al., 2022), exploring efficient means of augmenting biomedical PLMs with a domain-specific memory, avoiding expanding computations significantly. With BIOREADER, the biomedical knowledge is not necessary to be implicitly stored in model parameters but is explicitly acquired in a plug-and-play manner, leading to great scalability. Mechanically, BIOREADER is a novel encoder-decoder model based on T5 (Raffel et al., 2020) and RETRO (Borgeaud et al., 2021), with a frozen neural retriever. It splits the input sequence into chunks and autoregressively retrieves scientific text semantically similar to the previous fragment. In this way, it expands the input prompt context for better-predicting tokens in the current chunk thanks to a cross-attention mechanism.

Our main contributions are the following:

1. We devise BIOREADER, a novel text-to-text biomedical language model fusing memory retrieval and generative components (§3).
2. We advance biomedical NLP research, pushing the state-of-the-art in several knowledge-intensive and non-knowledge-intensive tasks via fine-tuning (§4 and §5), outperforming previous methods by a significant margin with up to 3x fewer parameters. We extensively prove the contribution of each module with ablation studies.
3. We show that BIOREADER can be improved at evaluation time by updating the knowledge base and the number of retrieved neighbors without retraining (§5), also offering qualitative benefits in terms of interpretability.

2 Related Work

We first give a bird's-eye view of existing work on biomedical language modeling and retrieval-enhanced neural networks (see Table 1).

Biomedical Language Models Transformer-based PLMs have become the first choice for any task in biomedical NLP, counting 40+ models proposed in just two years (Kalyan et al., 2022). Domain adaption milestones include contributions like BIOBERT (Lee et al., 2020), PUBMEDBERT (Gu et al., 2022), BIOMEGATRON (Shin et al., 2020), and SCIFIVE (Phan et al., 2021), as well as knowledge-enhanced encoders (Liu et al., 2021). Crucially, existing models still struggle to encapsulate high amounts of biomedical knowledge in their parameters (Frisoni et al., 2020b; Meng et al., 2022). As far as we can tell, we are the first to inspect retrieval-enhanced biomedical text generation, where open-book language models are ostensibly scarce.

Retrieval-Augmented Neural Networks The retrieval and knowledge grounding paradigms have lately attracted many computational linguists, aiming to design modular architectures capable of separating memory storage and computational pro-

Model	Granularity	Retriever training	Retrieval integration	Unsupervised Retriever	Retrieval Source	Task(s)
kNN-LM Khandelwal et al. (2020)	Token	Frozen (Transformer)	Add to probs	✓	Wikipedia, Books	LM
SPALM Yogatama et al. (2021a)	Token	Frozen (Transformer)	Gated logits	✓	Wikipedia	OpenQA
DPR Karpukhin et al. (2020)	Prompt	Contrastive proxy	Extractive QA		Wikipedia	OpenQA
REALM Guu et al. (2020)	Prompt	End-to-End	Prepend to prompt	✓	Wikipedia	OpenQA
RAG Lewis et al. (2020b)	Prompt	Fine-tuned DPR	Cross-attention (concatenation)		Wikipedia	OpenQA, QG, FV
FID Izacard and Grave (2021)	Prompt	Fine-tuned DPR	Cross-attention		Wikipedia	OpenQA
EMDR^2 Sachan et al. (2021)	Prompt	End-to-end	Cross-attention	✓	Wikipedia	OpenQA
RETRO Borgeaud et al. (2021)	Chunk	Frozen (BERT)	Chunked cross-attention	✓	Web, Books, News, Wikipedia, GitHub	OpenQA
BIOREADER (ours)	Chunk	Frozen (CONTRIEVER)	Chunked cross-attention	✓	PubMed†	NER, RE, DC, NLI, QA, OpenQA

Table 1: Comparison of BIOREADER with existing retrieval approaches. LM = language modeling, QG = question generation, FV = fact verification, NER = named entity recognition, RE = relation extraction, DC = document classification, NLI = natural language inference, (Open)QA = (open-domain) question answering. † highlights retrieval sources that are different from training data.

cessing. A popular strategy (Chen et al., 2017; Yang et al., 2019; Nie et al., 2019) relies on collecting passages employing untrained sparse-vector retrieval methods with inverted index matching, such as TF-IDF (Domeniconi et al., 2015) and BM25 (Robertson and Zaragoza, 2009), eventually improved by re-ranking (Wang et al., 2018). Other works identify relevant neighbors through latent topic modeling (Wei and Croft, 2006; Domeniconi et al., 2016c), edit-distance (Zhang et al., 2018; Gu et al., 2018), or algebraic methods (Domeniconi et al., 2016b; Frisoni et al., 2020a; Frisoni and Moro, 2020; Frisoni et al., 2020c). The source database may also be structured (Ahn et al., 2016; Yasunaga et al., 2021), and graphs may serve as a foundation for non-parametric retrievers guided by entity links to find chains of evidence documents (Asai et al., 2020). With the success of deep learning, retrieving systems have partly switched to dense learned semantic representations and distances in embedding spaces, mostly involving BERT-based architectures. Encodings can be pre-computed and indexed offline for greater efficiency and scalability (Grave et al., 2017; Seo et al., 2018; Khandelwal et al., 2020; Yogatama et al., 2021b; Borgeaud et al., 2021). For instance, kNN-LM (Khandelwal et al., 2020) extends a PLM by linearly interpolating its next token distribution with a k-nearest neighbors mechanism, without incorporating the retrieval process into the training pipeline. In this sense, k-nearest-neighbors have been largely investigated in NLP tasks (Domeniconi et al., 2014b). Retrieval metrics may also be learned from data in a task-dependent way instead of relying on pre-existing PLMs. Following this vision, DPR (Karpukhin et al., 2020) fine-tunes two BERT models utilizing a contrastive loss to align labeled query and key embeddings, thereby working with passage (or chunk) granularities. RAG (Lewis et al., 2020b) and FID (Izacard and Grave, 2021) build upon DPR to incorporate retrieval into seq2seq models. Nevertheless, source-target pairs preclude the use of abundant unlabeled data. Still, except for RAG, the retriever network is trained in isolation from the downstream task. To overcome this potential issue, end-to-end approaches have been recently proposed, including REALM (Guu et al., 2020) and EMDR^2 (Sachan et al., 2021), also exploiting unsupervised pre-training objectives to reward informative retrieval, like perplexity maximization and inverse cloze task as in SPALM (Lee et al., 2019). On the flip side, joint retriever-reader learning comes with the extra complexity of back-propagating while making queries on an entire corpus and periodically updating the embedding table, severely limiting scalability. In open-ended text generation, BLENDERBOT 2.0 (Komeili et al., 2022) and WEBGPT (Nakano et al., 2021) learn to make contextualized internet search queries to leverage up-to-the-minute information, but need huge amounts of annotations. Most pertinent to our work is RETRO (Borgeaud et al., 2021), a flexi-

ble autoregressive PLM conditioned on document chunks retrieved from trillions of tokens, significantly outperforming GPT-3 (Brown et al., 2020) with an order of magnitude fewer parameters. Like kNN-LM, SPALM and RETRO, BIOREADER uses frozen retrieval representations to easily accommodate the biomedical literature evolution, not requiring retraining in the event of a knowledge base change. Inspired by the promises of RETRO, whose code and models have not been released, BIOREADER processes arbitrary text sequences by reasoning at a sub-sequence level and retrieving different biomedical passages for the different chunks of a sequence, thus allowing for repeated retrieval during text generation. As suggested by the latest research thread for future directions (Borgeaud et al., 2021), we adopt a brand-new architecture derived from T5 to depend more on the encoder output at inference time. T5 fine-tuning has become a staple of natural language generation, marking off the prominent technique of many tasks (Paolini et al., 2021; Geng et al., 2022; Frisoni et al., 2022a), characterized by better grammatical correctness and transfer learning. In contrast to the majority of works that either interpolate output probabilities (Khandelwal et al., 2020) or use input concatenation (Yogatama et al., 2021a; Lewis et al., 2020b; Guu et al., 2020) to combine retrieved documents, BIOREADER separately encodes input prompts and neighbors, then assembled with a chunked cross-attention. Our work contrasts previous efforts on building T5 closed-domain models without access to any external context for knowledge-intensive tasks (Roberts et al., 2020).

Multi-Task Retrieval Prior work has shown that retrieval improves performance across various NLP tasks—especially extractive (Guu et al., 2020; Lee et al., 2019)—when considered in isolation. Such downstream tasks include open-domain question answering (Chen et al., 2017), fact-checking (Thorne et al., 2018), machine translation (Zhang et al., 2018), multi-document summarization (Moro et al., 2022), and data-to-text (Su et al., 2021). As first proposed by RAG, we demonstrate that a single retrieval-based seq2seq architecture can outperform several abstract biomedical benchmarks, also in multi-task settings. Further, we evince the advantages of merging retrieval and generative components also in non-knowledge-intensive tasks.

3 Method

Motivated by a recent stream of architectural contributions and training modalities (Guo et al., 2021; Sanh et al., 2021), our proposed text-to-text model BIOREADER extends T5 with the nimble ability to generate a sequence conditioned by a collection of passages retrieved from a specialized datastore with several millions of biomedical evidence tokens, other than the input. Coarse-grained retrieval of contiguous token chunks allows us to retain storage and computation requirements. The unique characteristics of the biomedical text and the related design choices are summarized in §A. Figure 2 sketches the overall framework.

Input segmentation We split the tokenized input X (max-length n) into a sequence of l chunks of size $m{=}\frac{n}{l}$. We use $n{=}512$ and $m{=}16$. Our approach uses retrieval as a way to augment input examples at the granularity of small chunks.

3.1 Retrieving scientific evidence

Evidence Datastore We use a retrieval pool different from the training corpus, which is suitable for domain adaptation and knowledge update (Li et al., 2022). Our database $\mathcal{D}$ is an external key-value store queried during inference. We derive $\mathcal{D}$ from PubMed-RCT (Dernoncourt and Lee, 2017), consisting of ≈200K English abstracts of randomized controlled trials (RCTs) from the 2016 MEDLINE/PubMed Baseline Database. We focus on RCTs as they are commonly considered the best source of medical evidence (Dickersin and Li, 2015). Let $f(\cdot)$ be the function that maps a textual context to a fixed-length vector representation given by a frozen non-causal bi-directional encoder. Within $\mathcal{D}$, each value consists of two contiguous unlabeled chunks $[N, F]$, where N is the *neighbor* chunk and F is its *continuation* in the original abstract. The corresponding key is $f(N)$, pre-computed to enable online database modification and retrieval from huge amounts of data. Using both N and F as retrieved tokens helps increase model performance (see §D). We implement $f(\cdot)$ using CONTRIEVER (Izacard et al., 2021), a dual-encoder architecture based on BERT-base uncased (WordPiece tokenizer) and trained with the MoCo contrastive loss (He et al., 2020) for unsupervised retrieval, where queries and documents are encoded independently using the same model. Average pooling is applied over the outputs of the last layer to obtain one-vector representations. Our

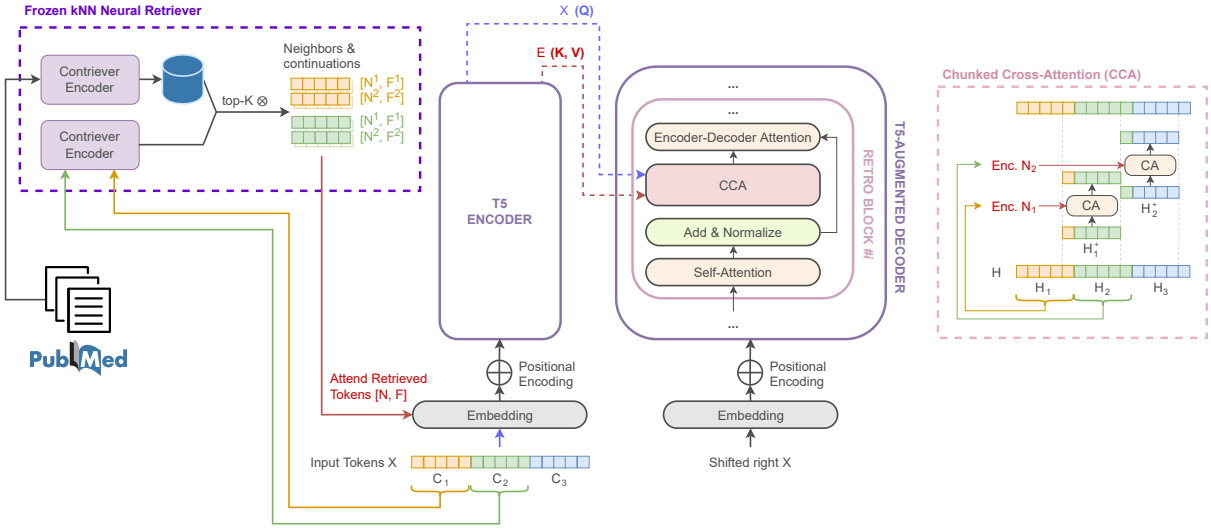

Figure 2: An illustration of the BIOREADER architecture. *Left:* simplified version where a sequence X of length n=15 is divided into l=3 chunks of size m=5. For each chunk, we retrieve k=2 scientific evidence neighbors of r=10 tokens each (including continuations). The current input prompt X and the fetched tokens are given as input to our encoder-decoder architecture based on T5. The fusion of their learned representations is done in the decoder via chunked-cross attention (CCA). *Right:* autoregressive CCA interaction details.

document index can be seen as a large external human-readable/writable memory for PLMs to attend to, on a par with memory networks (Moro et al., 2018).

Nearest neighbor retrieval For each input chunk C_u with $u \in \{1, l\}$, we select its top k most similar documents using the dot product $d(C_u, N) = f(C_u) \otimes f(N)$ (empirically better than L_2 distance according to preliminary experiments). Time- and memory-efficient retrieval is performed using FAISS (Johnson et al., 2021), an open source library for approximate nearest neighbor search in high dimensional spaces (sub-linear memory access). We denote retrieved token-values as $\text{RET}(C_u) = ([N^1, F^1], \ldots, [N^k, F^k])$. A length of 16 is used for both N^j and F^j, thus $\text{RET}(C_u)$ has a $k \times r$ shape, with r=32.

3.2 Model architecture

BIOREADER relies on an extended T5 architecture, receiving X chunks as input. Concretely, we keep the T5 encoder unchanged, while we interleave the RETRO-blocks proposed by Borgeaud et al. 2021 and standard T5-blocks in the decoder—design choices are motivated in §D. Symbolizing intermediate input activations by $H \in \mathbb{R}^{n \times d}$, RETRO-blocks incorporate information also from the encoded neighbors E—for which we take the T5-encoded $\text{RET}(C_u)$ tokens, already supplied with positional information. RETRO-blocks compose three different residual operators with signature $\mathbb{R}^{n \times d} \to \mathbb{R}^{n \times d}$: a fully connected layer FFW, a standard self-attention layer ATT, and a chunked cross-attention layer $\text{CCA}(\cdot, E)$.

$$\text{RETRO}(H, E) = \text{FFW}(\text{CCA}(\text{ATT}(H), E)) \quad (1)$$

$$\text{T5}(H) = \text{FFW}(\text{ATT}(H)) \quad (2)$$

The hyperparameter $P \subseteq [1, L]$ determines at which layers a RETRO-block is placed in the decoder stack; referring to T5-base (L=12, d=768), we use P={9, 12}. In these points, the neighbor encodings and the *trainable* BIOREADER input encodings are merged with CCA, replacing the original encoder outputs; the resulting representation is then used for the Encoder-Decoder Attention. Our final configuration consists of 229.5M parameters.

Chunked cross-attention Following Borgeaud et al. 2021, we use chunked-cross attention (CCA), an autoregressive operator that incorporates the retrieved literature evidence into the model. To compute it, a given activation H is first divided into $l-1$ chunks, shifting the tokens composing each chunk of one position to the left $(H_u^+ = (h_{u\,m+i-1})_{i \in [1,m]} \in \mathbb{R}^{m \times d})_{u \in [1, l-1]}$. Note that H_u^+ holds the embeddings of the last token in chunk C_u and of the first $m-1$ tokens in C_{u+1}. The cross-attention between H_u^+ and E_u is then calculated across time and neighbors simultaneously.

This means that each input chunk attends only to the neighbors of the preceding chunk; autoregression is ensured by the one-token overlap (i.e., the dependencies over previous neighbors are propagated via self-attention operations). The activations of the i^{th} token in the u^{th} chunk therefore potentially depend upon the set of all previous neighbors and continuations $\text{RET}(C_{u'})_{u'<u}$. A graphical example is reported in §F.

Retrieval-enhanced autoregressive decoding Token likelihoods during decoding (i-th token, u-th chunk)—with BIOREADER parameterized by θ—depend on both previously seen tokens $PX_{u,i}$ and antecedent-chunk neighbors PN_u recovered by $\mathcal{D}$, nailing down a retrieval-enhanced sequence log-likelihood, following Borgeaud et al. 2021.

$$PX_{u,i} = x_{(u-1)\,m+i} | (x_j)_{j<(u-1)\,m+i}, \quad (3)$$

$$PN_u = (\text{RET}(C_{u'}))_{u'<u}, \quad (4)$$

$$L(X|\theta, \mathcal{D}) = \sum_{u=1}^{l} \sum_{i=1}^{m} \ell_\theta(PX_{u,i}, PN_u). \quad (5)$$

We can select the next token by directly sampling from the $\mathcal{D}$-conditioned distribution with log-probability ℓ. Specifically, we adopt the standard T5 greedy decoding approach and set $\text{RET}(C_1)=\varnothing$, namely, the likelihood of tokens from the first chunk does not depend on any neighbor.

Differences compared to RETRO Differently from RETRO, BIOREADER is characterized by an original T5 skeleton (instead of GPT and a decoder-modulated transformer encoder), different architectural and retrieval pool design choices, a biomedical-specific model (also for encoding retrieved tokens), a non-training-based retrieval pool, and a dual-encoder retrieval module trained specifically for fetching short documents given a query.

4 Experimental Setup

Implementation details, computing infrastructure, and experiment hyperparameters are described in §B.1 and §B.2.

4.1 Training

Pre-training corpora In biomedical PLMs, there are two main sources for pre-training corpora: PubMed abstracts and PMC articles. As demonstrated by prior work (Phan et al., 2021; Gu et al., 2022), training on both corpora surprisingly leads to a slight degradation in performance compared to solely training on PubMed abstracts. Consequently, we operate on a cleaned and masked version of the PubMed database (>32M abstracts)[2].

Pre-training and fine-tuning setup We first initialize BIOREADER's T5-blocks by loading the pre-trained weights of SCIFIVE(PubMed)-base, a state-of-the-art T5 model pre-trained on large biomedical corpora. Referring to the findings of Borgeaud et al. 2021[3], we freeze all the pre-trained weights and train only the new CCA parameters (less than 5% of total weights) with span-based mask learning. Thus, the original SCIFIVE performance is precisely maintained when BIOREADER language modeling is evaluated without retrieval. We pre-train CCA layers by using only ≈3% of pre-training corpora instances (≈982K)—a design choice supported by Borgeaud et al. 2021 as well. Consistently with the authors of RETRO, we find that allowing the entire model to resume training at this stage ends in performance worsening. Since the size of PubMed abstracts rarely exceeds 512 tokens (Yuan et al., 2022), we truncate all the input texts to a 512 maximum length for the sake of pre-training efficiency. Subsequently, we fine-tune all the layers on the supervised target tasks (see §4.2), also making use of a task-specific prefix to let the model know the requested transformation for each input. During training, we retrieve from $\mathcal{D}$ with 9 neighbors; we raise and reduce the k value in the evaluation phase (§D).

Objective BIOREADER is trained with a maximum likelihood objective using teacher forcing (Raffel et al., 2020) for all tasks so as to unlock multi-task learning.

4.2 Downstream Benchmark Tasks

We fine-tune BIOREADER on 18 widespread NLP human-annotated biomedical datasets for 6 downstream task categories. Evaluation datasets mostly come from BLURB (Gu et al., 2022), a broad-coverage benchmark for PubMed-based biomedical NLP applications, tracking progress by the community. Unless otherwise specified, we follow the same preprocessing techniques and train/dev/test sets as Phan et al. 2021. Descriptive statistics and preprocessing details are in §C.

[2] gs://scifive/pretrain/pubmed_cleaned

[3] Uniquely tuning the new weights of CCA-augmented PLMs attains results close to full training from scratch, quickly surpassing the performance of baseline models.

- **Named entity recognition (NER)**. Locate and classify named biomedical entities using IOB tagging (Ramshaw and Marcus, 1995). We take into account 7 influential datasets: NCBI-disease (Dogan et al., 2014), BC5CDR-disease (Li et al., 2016), BC5CDR-chemical (Li et al., 2016), BC4CHEMD (Krallinger et al., 2015), BC2GM (Smith et al., 2008), JNLPBA (Collier and Kim, 2004), and Species800 (Pafilis et al., 2013). For all NER tasks, we evaluate results with the entity-level F1-score, ensuring fairness with the other baselines. We italicize that the entity-level evaluation does not count the partial prediction of an entity as true (if the entity has more than one token), tending to show lower scores than plain F1.
- **Relation extraction (RE)**. Detect and classify semantic relationships involving biomedical entities. We test on CHEMPROT (Dogan et al., 2019) and DDI (Herrero-Zazo et al., 2013) for chemical-protein and disease-disease interactions, respectively. We evaluate the F1-score of each class in the two datasets.
- **Natural language inference (NLI)**. Determine the validity of a hypothesis (i.e., true or false). We utilize the MedNLI dataset from MIMIC-III (Romanov and Shivade, 2018) with an accuracy-based evaluation.
- **Document classification (DC)**. Assign a text document to a predetermined category. We consider the HoC dataset (Baker et al., 2016), judging the F1-score on the sample average.
- **Question answering (QA)**. Find an answer to a question from a gold context snippet. We take factoid questions from BioASQ 4b, 5b, and 6b challenges (Tsatsaronis et al., 2015).
- **Open-domain QA (OpenQA)**. Answer natural questions without relying on any specified context paragraph. OpenQA is a common knowledge-intensive testbed for retrieve-then-generate models. We refer to MedQA-USMLE (Jin et al., 2021), a 4-way multiple choice QA benchmark entailing biomedical and clinical knowledge, where the questions originate from practice tests for the United States Medical License Exams (USMLE). We preprocess the dataset by treating questions and correct answers as input-output text pairs. In all cases where a question refers to a specific set of answers (e.g., "which of the following..."), we append the possible choices to the input text.

For QA and OpenQA, we observe standard conventions and evaluate the predicted free text with the Exact Match metric, as initiated by Rajpurkar et al. 2016. A generated answer (lenient for QA) is considered correct if it matches any reference answer after normalization (i.e., lowercasing and removal of articles, punctuation, and duplicated whitespace).

Comparison We head-to-head compare BIOREADER to representative closed-book PLMs, encompassing prevalent BERT-based models requiring task-specific architectural choices (prediction heads) and more flexible encoder-decoder generative models. The first category includes BIOBERT (Lee et al., 2020), SCIBERT (Beltagy et al., 2019), BLUEBERT (Peng et al., 2019), CLINICALBERT (Alsentzer et al., 2019), PUBMEDBERT (Gu et al., 2022), PUBMEDELECTRA (Tinn et al., 2021), BIOLINKBERT (Yasunaga et al., 2022), and BIOMEGATRON (Shin et al., 2020). The second, T5 (Raffel et al., 2020) and SCIFIVE (Phan et al., 2021). Please note that BIOMEGATRON authors evaluate NER performance by labeling sub-tokens separately, except for the NCBI-disease dataset, where they observe better results with whole-entity labeling. We also mention BIOROBERTA (Lewis et al., 2020a) and BIOBART (Yuan et al., 2022), which are not included due to the impossibility of replicating them on the BLURB dataset splits.

4.3 Qualitative Analysis

Online evidence datastore update Facts memorized within traditional PLMs are opaque and stuck in time at the point of training (Lazaridou et al., 2021). Such static knowledge fails to cope with the dynamic state of the biomedical world, where more than 3 papers are registered per minute (Frisoni et al., 2021). With BIOREADER, we can control what the model knows by swapping out or integrating the documents it uses for knowledge retrieval. We test this behavior by adding to $\mathcal{D}$ the abstracts of 10 recent RCTs on COVID-19[4], checking for factual answers to target open questions *without retraining*. We consider the OpenQA-tuned model, assessed on two relevant questions created by us

[4]We select RCTs by employing "review covid19 symptoms" (x3), "review covid19 prevention" (x3), and "review covid19" (x4) as keywords on the PubMed search engine.

according to new RCT contents.

Question answering human evaluation In QA and OpenQA benchmarks, BIOREADER outputs full-sentence answers that often do not correspond to the ground truth but continue to be semantically correct. We hypothesize that exact matching underestimates our model performance. For this reason, we hire three expert human annotators proficient in English and with biomedical competencies to manually scrutinize model predictions, randomly sampling 120 test set instances (30 from each BioASQ dataset and 30 from MedQA-USMLE). We ask the graders to (i) binary label the scientific accuracy (factual correctness) of the generated answers, (ii) assess language fluency on a 3-point Likert scale from 1 (worst) to 3 (best). Human evaluation is conducted on SCIFIVE and BIOREADER outputs (presented in random order) to inspect the retrieval-augmentation contribution.

5 Results

Table 2 and Table 3 showcase our main results. Our scores come from the checkpoint with the lowest loss and the best k discovered at evaluation time.

We push the state-of-the-art on 2/7 NER, 1/2 RE, 1/1 DC, and 3/3 QA, staying highly competitive in all the other cases. We beat SCIFIVE-large (3x our size) on 5 different tasks, while, in the majority of cases, we considerably outperform models which have a comparable number of parameters to ours. Overall performances testify to consistent retrieval effectiveness. Predicting tokens with the aid of relevant human-written references alleviates the difficulty of text generation. As expected, we notice that NER, RE, and NLI are the tasks where BIOREADER contributes less. Naturally, all these translations are strictly related to the provided inputs and hardly take advantage of additional external context, which often acts as noise.

The strength of BIOREADER is accentuated when limited training data is available; we point up that our biomedical benchmarks only have few thousand annotated instances (§C).

Unexpectedly, the adoption of in-domain vocabularies appears to be non-correlated with higher downstream task scores.

From Table 3, we verify our conjecture about the insufficiency of Exact Match as an OpenQA evaluation metric. Expert assessment results are significantly higher than the automatic ones. The average Kendall's coefficient ($[-1, 1]$ bound) among all evaluators' inter-rater agreement is 0.91. We recognize many cases where the predicted answer is correct though different from the ground truth (e.g., "pd-1" vs. "programmed cell death 1", "olfactory groove meningioma" vs. "meningioma"). Our qualitative analysis suggests that neighbors help the model to produce not only more syntactically fluent but also more factually correct outputs.

Although not retrained, BIOREADER adapts correctly to unseen questions on the COVID-19 literature in "zero-shot datastore" settings (Table 4). This suggests that it learns to use world information independently of the information itself. Input-output examples, accompanied by their retrieved neighbors, are exhibited in §F.

Replicating our solution (see §B) only asks for CCA calibration and task-specific model fine-tuning, ultimately saving a vast amount of computation and memory. Our model can be handled on a single GPU machine, while a fully end-to-end retriever generally demands industry-scale computational resources for training (Seo et al., 2019).

6 Conclusions

In this paper, we introduce BIOREADER, a new state-of-the-art semi-parametric biomedical language model steered by literature passages retrieved from explicit memory. Our experimental results show that augmenting the generation process by accessing scientific repositories during training and evaluation induces performance gains greater than raw parameter scaling, both on knowledge-intensive and non-knowledge-intensive tasks. Not only do we provide a way of handling the opaqueness of large language models, but we also prove that updating the datastore helps the model with domain adaption without retraining, a property of paramount importance for rapidly evolving domains like biomedicine. Future work should aim to integrate a differentiable write-to-memory operator (Wu et al., 2022), structured retrieval databases (e.g., multi-relational graphs from semantic parsing, symbolic knowledge graphs), long text-to-text tasks (Guo et al., 2021; Moro and Ragazzi, 2022), knowledge-augmented self-alignment pre-training to rewire the space before retrieval (Liu et al., 2021), and the evaluation of distributed and parallel learning approaches to scale to big data repositories (Cerroni et al., 2013).

Model	#params	In-Domain Vocabulary	NER (F1)							RE (F1)		DC (F1*)	NLI (Acc.)
			NCBI disease	BC5CDR disease	BC5CDR chemical	BC4CHEMD	BC2GM	JNLPBA	Species-800	ChemProt	DDI	HoC	MedNLI
BioBERT†,‡	110M	✓	**89.71**	87.15	93.47	92.36	84.72	77.49	74.06	76.46	80.88	81.54	—
SciBERT‡	110M	✓	88.25	84.70	92.51	—	83.36	78.51	—	75.00	81.22	81.16	—
BlueBERT-base‡	110M	×	88.04	83.69	91.19	—	81.87	77.71	—	71.46	77.78	80.48	—
ClinicalBERT‡	110M	×	86.32	83.04	90.80	—	81.71	78.07	—	72.04	78.20	80.74	—
PubMedBERT-base‡	110M	✓	87.82	85.62	93.33	—	84.52	79.10	—	77.24	—	—	—
PubMedBERT-large§	340M	✓	88.25	85.77	93.22	—	84.72	79.44	—	78.77	82.39	82.57	—
PubMedElectra-base§	110M	✓	87.68	84.99	93.19	—	83.79	78.60	—	76.54	80.58	81.45	—
PubMedElectra-large§	340M	✓	87.93	84.82	92.90	—	83.87	78.77	—	76.80	78.92	82.37	—
BioLinkBERT-base‖	110M	×	88.18	86.10	93.75	—	84.90	79.03	—	77.57	82.72	84.35	—
BioLinkBERT-large‖	340M	×	88.76	86.39	94.04	—	**85.18**	**80.06**	—	79.98	83.35	84.87	—
BioMegatron¶	345M	✓	87.10	**88.50**	92.90	—	—	—	—	77.00	—	—	—
T5-base†	220M	×	88.54	86.83	93.61	89.73	82.29	74.56	74.32	84.82	82.04	85.22	83.90
T5-large†	770M	×	88.78	86.31	94.22	89.96	82.36	75.83	74.66	85.41	83.35	85.68	83.80
SciFive-base†	220M	×	87.96	87.44	94.35	92.02	83.92	75.60	76.55	**88.83**	83.15	85.89	85.30
SciFive-large†	770M	×	89.17	86.98	**94.66**	91.96	83.60	76.08	75.50	87.88	83.67	86.36	**86.36**
BioReader (ours)	229.5M	×	88.90	87.62	94.43	**92.81**	84.77	77.82	**77.44**	88.16	**84.34**	**87.78**	85.76

Table 2: Test results on NER, RE, DC, and NLI after fine-tuning. F1* is F1 on sample average. **Bold** and underline denote the best and second best scores; the gradient of green indicates our improvement compared to the previous state-of-the-art (the deeper, the more). †, ‡, §, ‖ and ¶ baseline results (correctly replicated except for PubMedBERT-large, PubMedElectra, BioMegatron) are from Phan et al. (2021), Gu et al. (2022), Tinn et al. (2021), Yasunaga et al. (2022), and Shin et al. (2020), respectively. "—" denotes no results are available.

Model	# params	In-Domain Vocabulary	Automatic Evaluation				Human Evaluation				
			QA			OpenQA	QA			OpenQA	Fluency
			BioAsq 4b	BioAsq 5b	BioAsq 6b	MedQA-USMLE	BioAsq 4b	BioAsq 5b	BioAsq 6b	MedQA-USMLE	(Avg)
BioLinkBERT-base	110M	×	—	—	—	40.00	—	—	—	—	—
BioLinkBERT-large	340M	×	—	—	—	**44.60**	—	—	—	—	—
SciFive-base	220M	×	60.80	59.53	55.56	34.57	79.98	80.02	70.05	38.03	2.49
SciFive-large	770M	×	62.98	61.67	61.74	35.12	80.23	80.12	71.54	39.78	2.65
BioReader (ours)	229.5M	×	**64.13**	**62.02**	**62.18**	42.96	**82.12**	**81.88**	**73.35**	**48.57**	**2.86**

Table 3: Exact Match accuracy (left) and human-evaluated scientific accuracy (right) on QA and OpenQA tasks. **Bold** and underline denote the best and second best scores; our relative human evaluation improvement compared to the baseline is picked out with green gradients (the deeper, the more).

Question	BioReader w/ $\mathcal{D}$		BioReader w/ $\mathcal{D}'$	
medqa: question*: January 2020. A 69-year-old Chinese man comes to the physician with fever, tiredness, cough, dyspnoea, and severe respiratory issues. The clinical picture suggests an infectious disease. What is the most likely diagnosis?	✗	bronchiolitis	✓	COVID-19
medqa: question*: Coronaviruses are viruses that can cause illnesses in humans, including severe respiratory disease and even death. Corona disease-19 virus (COVID-19) spread and caused a pandemic that affected people all over the world. As COVID-19 cases continue to rise globally, which are the most effective options to prevent contamination and infection transmission?	✗	disinfect the respiratory tract	✓	vaccinate against COVID-19

Table 4: Answers generated by BioReader to context-free COVID-19 questions before ($\mathcal{D}$) and after ($\mathcal{D}'$) integrating SARS-CoV-2 evidence into the datastore.

7 Ethical Considerations

The language model's ability to make the most of pre-existing domain knowledge could have potential ramifications for society, especially in healthcare contexts. From an application perspective, researchers need better NLP tools to skim the biomedical literature efficiently. Grounding in real factual evidence (in this case PubMed's RCTs) reduces hallucination phenomena and offers more control and interpretability. Users could endow BioReader with a sizeable medical index and ask it open-domain questions to avoid reading thousands of publications. Analogously, they could classify documents or perform structured prediction with a broader and up-to-date vision, going beyond the information provided (in a commonsense fashion) and taking advantage of similarities between tasks thanks to multi-task learning. By including a retrieval method, BioReader remains relatively small in size: plenty of users can deploy it on affordable GPUs and tweak it as needed. Furthermore, the applications of this paper are beyond the biomedical domain only, being suitable for targeting (i) limited resource domains, (ii) out-

of-distribution issues in downstream tasks (Parmar et al., 2022), and (iii) domain-adaption with limited fine-tuning datasets.

With these benefits also come potential downsides. Indeed, any external knowledge source will probably never be entirely factual, coherent, and completely devoid of bias, particularly on large scales. We urge the users to undertake the necessary quality-assurance testing to understand the presence of such issues and evaluate how much they impact the model. On the other side, one advantage of using an explicit external memory is that the latter can be easily cleared, edited, or retroactively filtered. The same is not true of siloed knowledge in traditional PLMs. Like any large language model, BIOREADER could be the subject of concern about its malicious use, although arguably to a lesser extent. For example, it might be used to automate the production of faked or misleading content, which could be critical in sensitive healthcare domains.

We honor and support the ACL code of Ethics. All pre-trained models and corpora used in this work are publicly available.

8 Limitations

Chunks may contain only partial information about biomedical evidence, with the risk of generating incomplete or nonfactual text. Also, multiple chunks can refer to the same fact; even if such retrieved passages have the prospect of complementing each other, they can cause repetitions or contradictions. Managing contradictions—which are natural in the scientific evolution of a field over time—is precisely one of the main future research directions we envisage. Context-sensitive chunks may also be considered when building the knowledge base to avoid splitting within word or entity boundaries, which is especially risky in biomedicine.

We take only abstracts for constructing our knowledge base: future work should explore massive-scale full-texts and their implications with respect to the results presented in this paper. Indeed, BIOREADER performances are capped by the contained topic coverage of the selected datastore.

We believe that a quantitative assessment of the link between the datastore modifications and their effect on model predictions is imperative, drawing attention to the need for new benchmarks.

Additionally, given the high memory consumption and large space on disk potentially required by FAISS indexes, we suggest the reader adopt a Binary Passage Retriever model (Yamada et al., 2021), which reduces the index size without losing too much in performance.

Finally, our model backbone (SCIFIVE) may be undertrained, reckoning on significantly fewer computational resources (i.e., a single TPUv2-8) than the ones employed for the original T5 and baselines like PUBMEDBERT. We show promising results in constrained settings imposed by our GPU limitations, striving to make our work as reproducible as possible and leaving the possibility of adapting it to more performing hardware. We encourage future researchers to replicate our paper and unveil its real potential with well-trained seq2seq models such as BIOBART (Yuan et al., 2022), pre-trained on biomedical corpora with 16 40GB A100 GPUs for 7 days. Alternatively, we suggest bettering the pre-train of bio-T5 models, possibly using the DeepNarrow strategy proposed by Tay et al. 2021, which reduces costs by training 50% fewer parameters and being 40% faster.

We hope that our work may trigger the community toward the development of new open-book biomedical models and datasets, lowering the entry barrier and helping to accelerate progress in this vitally important field for positive societal and human impact.

Acknowledgements

We would like to thank all the anonymous reviewers for their constructive, detailed, and valuable comments.

References

Sungjin Ahn, Heeyoul Choi, Tanel Pärnamaa, and Yoshua Bengio. 2016. A neural knowledge language model. *CoRR*, abs/1608.00318.

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2020. Learning to retrieve reasoning paths over wikipedia graph for question answering. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Simon Baker, Ilona Silins, Yufan Guo, Imran Ali, Johan Högberg, Ulla Stenius, and Anna Korhonen. 2016. Automatic semantic classification of scientific literature according to the hallmarks of cancer. *Bioinform.*, 32(3):432–440.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. 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 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2021. Improving language models by retrieving from trillions of tokens. *CoRR*, abs/2112.04426.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom B. Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*, pages 2633–2650. USENIX Association.

Walter Cerroni, Gianluca Moro, Tommaso Pirini, and Marco Ramilli. 2013. Peer-to-peer data mining classifiers for decentralized detection of network attacks. In *Twenty-Fourth Australasian Database Conference, ADC 2013, Adelaide, Australia, February 2013*, volume 137 of *CRPIT*, pages 101–108. Australian Computer Society.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada. Association for Computational Linguistics.

Nigel Collier and Jin-Dong Kim. 2004. Introduction to the bio-entity recognition task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP)*, pages 73–78, Geneva, Switzerland. COLING.

Joe Davison, Joshua Feldman, and Alexander Rush. 2019. Commonsense knowledge mining from pretrained models. 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 1173–1178, Hong Kong, China. Association for Computational Linguistics.

Franck Dernoncourt and Ji Young Lee. 2017. PubMed 200k RCT: a dataset for sequential sentence classification in medical abstracts. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 308–313, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Pietro di Lena, Giacomo Domeniconi, Luciano Margara, and Gianluca Moro. 2015. GOTA: GO term annotation of biomedical literature. *BMC Bioinform.*, 16:346:1–346:13.

Kay Dickersin and Tianjing Li. 2015. Introduction to systematic review and meta-analysis. Coursera.

Rezarta Islamaj Dogan, Sun Kim, Andrew Chatr-aryamontri, Chih-Hsuan Wei, Donald C. Comeau, Rui Antunes, Sérgio Matos, Qingyu Chen, Aparna Elangovan, Nagesh C. Panyam, Karin Verspoor, Hongfang Liu, Yanshan Wang, Zhuang Liu, Berna Altinel, Zehra Melce Hüsünbeyi, Arzucan Özgür, Aris Fergadis, Chen-Kai Wang, Hong-Jie Dai, Tung Tran, Ramakanth Kavuluru, Ling Luo, Albert Steppi, Jinfeng Zhang, Jinchan Qu, and Zhiyong Lu. 2019. Overview of the biocreative VI precision medicine track: mining protein interactions and mutations for precision medicine. *Database J. Biol. Databases Curation*, 2019:bay147.

Rezarta Islamaj Dogan, Robert Leaman, and Zhiyong Lu. 2014. NCBI disease corpus: A resource for disease name recognition and concept normalization. *J. Biomed. Informatics*, 47:1–10.

G. Domeniconi, M. Masseroli, G. Moro, and P. Pinoli. 2014a. Discovering new gene functionalities from random perturbations of known gene ontological annotations. pages 107–116. INSTICC Press.

Giacomo Domeniconi, Marco Masseroli, Gianluca Moro, and Pietro Pinoli. 2016a. Cross-organism learning method to discover new gene functionalities. *Comput. Methods Programs Biomed.*, 126:20–34.

Giacomo Domeniconi, Gianluca Moro, Andrea Pagliarani, Karin Pasini, and Roberto Pasolini. 2016b. Job Recommendation from Semantic Similarity of LinkedIn Users' Skills. In *ICPRAM 2016*, pages 270–277. SciTePress.

Giacomo Domeniconi, Gianluca Moro, Roberto Pasolini, and Claudio Sartori. 2014b. Iterative Refining of Category Profiles for Nearest Centroid Cross-Domain Text Classification. In *IC3K 2014, Rome, Italy, October 21-24, 2014, Revised Selected Papers*, volume 553, pages 50–67. Springer.

Giacomo Domeniconi, Gianluca Moro, Roberto Pasolini, and Claudio Sartori. 2015. A Comparison of Term Weighting Schemes for Text Classification and Sentiment Analysis with a Supervised Variant of

tf.idf. In *DATA (Revised Selected Papers)*, volume 584, pages 39–58. Springer.

Giacomo Domeniconi, Konstantinos Semertzidis, Vanessa López, Elizabeth M. Daly, Spyros Kotoulas, and Gianluca Moro. 2016c. A novel method for unsupervised and supervised conversational message thread detection. In *DATA 2016 - Proc. 5th Int. Conf. Data Science, Technol. and Appl., Lisbon, Portugal, 24-26 July, 2016*, pages 43–54. SciTePress.

Giacomo Frisoni and Gianluca Moro. 2020. Phenomena explanation from text: Unsupervised learning of interpretable and statistically significant knowledge. In *Data Management Technologies and Applications - 9th International Conference, DATA 2020, Virtual Event, July 7-9, 2020, Revised Selected Papers*, volume 1446 of *Communications in Computer and Information Science*, pages 293–318. Springer.

Giacomo Frisoni, Gianluca Moro, and Lorenzo Balzani. 2022a. Text-to-text extraction and verbalization of biomedical event graphs. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2692–2710, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Giacomo Frisoni, Gianluca Moro, and Antonella Carbonaro. 2020a. Learning Interpretable and Statistically Significant Knowledge from Unlabeled Corpora of Social Text Messages: A Novel Methodology of Descriptive Text Mining. In *DATA 2020 - Proc. 9th Int. Conf. Data Science, Technol. and Appl.*, pages 121–134. SciTePress.

Giacomo Frisoni, Gianluca Moro, and Antonella Carbonaro. 2020b. Towards Rare Disease Knowledge Graph Learning from Social Posts of Patients. In *RiiForum*, pages 577–589. Springer.

Giacomo Frisoni, Gianluca Moro, and Antonella Carbonaro. 2020c. Unsupervised Descriptive Text Mining for Knowledge Graph Learning. In *IC3K 2020 - Proc. 12th Int. Joint Conf. Knowl. Discovery, Knowl. Eng. and Knowl. Manage.*, volume 1, pages 316–324. SciTePress.

Giacomo Frisoni, Gianluca Moro, and Antonella Carbonaro. 2021. A Survey on Event Extraction for Natural Language Understanding: Riding the Biomedical Literature Wave. *IEEE Access*, 9:160721–160757.

Giacomo Frisoni, Gianluca Moro, Giulio Carlassare, and Antonella Carbonaro. 2022b. Unsupervised event graph representation and similarity learning on biomedical literature. *Sensors*, 22(1):3.

Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2022. Recommendation as language processing (RLP): A unified pretrain, personalized prompt & predict paradigm (P5). *CoRR*, abs/2203.13366.

Edouard Grave, Armand Joulin, and Nicolas Usunier. 2017. Improving neural language models with a continuous cache. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Jiatao Gu, Yong Wang, Kyunghyun Cho, and Victor O. K. Li. 2018. Search engine guided neural machine translation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5133–5140. AAAI Press.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2022. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Heal.*, 3(1):2:1–2:23.

Mandy Guo, Joshua Ainslie, David C. Uthus, Santiago Ontañón, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2021. Longt5: Efficient text-to-text transformer for long sequences. *CoRR*, abs/2112.07916.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9726–9735. Computer Vision Foundation / IEEE.

María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. 2013. The DDI corpus: An annotated corpus with pharmacological substances and drug-drug interactions. *J. Biomed. Informatics*, 46(5):914–920.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Towards unsupervised dense information retrieval with contrastive learning. *CoRR*, abs/2112.09118.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, Online. Association for Computational Linguistics.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2021. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2021. Billion-scale similarity search with gpus. *IEEE Trans. Big Data*, 7(3):535–547.

Katikapalli Subramanyam Kalyan, Ajit Rajasekharan, and Sivanesan Sangeetha. 2022. AMMU: A survey of transformer-based biomedical pretrained language models. *J. Biomed. Informatics*, 126:103982.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020. Generalization through memorization: Nearest neighbor language models. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Mojtaba Komeili, Kurt Shuster, and Jason Weston. 2022. Internet-augmented dialogue generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8460–8478, Dublin, Ireland. Association for Computational Linguistics.

Martin Krallinger, Obdulia Rabal, Florian Leitner, Miguel Vazquez, David Salgado, Zhiyong Lu, Robert Leaman, Yanan Lu, Donghong Ji, Daniel M. Lowe, Roger A. Sayle, Riza Theresa Batista-Navarro, Rafal Rak, Torsten Huber, Tim Rocktäschel, Sérgio Matos, David Campos, Buzhou Tang, Hua Xu, Tsendsuren Munkhdalai, Keun Ho Ryu, S. V. Ramanan, P. Senthil Nathan, Slavko Zitnik, Marko Bajec, Lutz Weber, Matthias Irmer, Saber A. Akhondi, Jan A. Kors, Shuo Xu, Xin An, Utpal Kumar Sikdar, Asif Ekbal, Masaharu Yoshioka, Thaer M. Dieb, Miji Choi, Karin Verspoor, Madian Khabsa, C. Lee Giles, Hongfang Liu, Ravikumar Komandur Elayavilli, Andre Lamurias, Francisco M. Couto, Hong-Jie Dai, Richard Tzong-Han Tsai, Caglar Ata, Tolga Can, Anabel Usie, Rui Alves, Isabel Segura-Bedmar, Paloma Martínez, Julen Oyarzabal, and Alfonso Valencia. 2015. The CHEMDNER corpus of chemicals and drugs and its annotation principles. *J. Cheminformatics*, 7(S-1):S2.

Esther Landhuis. 2016. Scientific literature: Information overload. *Nature*, 535(7612):457–458.

Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Tomás Kociský, Sebastian Ruder, Dani Yogatama, Kris Cao, Susannah Young, and Phil Blunsom. 2021. Mind the gap: Assessing temporal generalization in neural language models. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 29348–29363.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020a. Pretrained language models for biomedical and clinical tasks: Understanding and extending the state-of-the-art. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 146–157, Online. Association for Computational Linguistics.

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020b. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Huayang Li, Yixuan Su, Deng Cai, Yan Wang, and Lemao Liu. 2022. A survey on retrieval-augmented text generation. *CoRR*, abs/2202.01110.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. 2016. Biocreative V CDR task corpus: a resource for chemical disease relation extraction. *Database J. Biol. Databases Curation*, 2016.

Fangyu Liu, Ehsan Shareghi, Zaiqiao Meng, Marco Basaldella, and Nigel Collier. 2021. Self-alignment pretraining for biomedical entity representations. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4228–4238, Online. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International*

Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net.

Jörn Lötsch, Dario Kringel, and Alfred Ultsch. 2021. Explainable artificial intelligence (XAI) in biomedicine: Making AI decisions trustworthy for physicians and patients. *BioMedInformatics*, 2(1):1–17.

Zaiqiao Meng, Fangyu Liu, Ehsan Shareghi, Yixuan Su, Charlotte Collins, and Nigel Collier. 2022. Rewire-then-probe: A contrastive recipe for probing biomedical knowledge of pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4798–4810, Dublin, Ireland. Association for Computational Linguistics.

Gianluca Moro, Andrea Pagliarani, Roberto Pasolini, and Claudio Sartori. 2018. Cross-domain & In-domain Sentiment Analysis with Memory-based Deep Neural Networks. In *IC3K 2018*, volume 1, pages 127–138. SciTePress.

Gianluca Moro and Luca Ragazzi. 2022. Semantic Self-Segmentation for Abstractive Summarization of Long Legal Documents in Low-Resource Regimes. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Virtual Event, February 22 - March 1, 2022*, pages 1–9. AAAI Press.

Gianluca Moro, Luca Ragazzi, Lorenzo Valgimigli, and Davide Freddi. 2022. Discriminative marginalized probabilistic neural method for multi-document summarization of medical literature. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 180–189, Dublin, Ireland. Association for Computational Linguistics.

Gianluca Moro and Lorenzo Valgimigli. 2021. Efficient self-supervised metric information retrieval: A bibliography based method applied to COVID literature. *Sensors*, 21(19).

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. 2021. Webgpt: Browser-assisted question-answering with human feedback. *CoRR*, abs/2112.09332.

Yixin Nie, Songhe Wang, and Mohit Bansal. 2019. Revealing the importance of semantic retrieval for machine reading at scale. 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 2553–2566, Hong Kong, China. Association for Computational Linguistics.

Evangelos Pafilis, Sune P Frankild, Lucia Fanini, Sarah Faulwetter, Christina Pavloudi, Aikaterini Vasileiadou, Christos Arvanitidis, and Lars Juhl Jensen. 2013. The species and organisms resources for fast and accurate identification of taxonomic names in text. *PloS one*, 8(6):e65390.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cícero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Mihir Parmar, Swaroop Mishra, Mor Geva, and Chitta Baral. 2022. Don't blame the annotator: Bias already starts in the annotation instructions. *CoRR*, abs/2205.00415.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language Models as Knowledge Bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Long N. Phan, James T. Anibal, Hieu Tran, Shaurya Chanana, Erol Bahadroglu, Alec Peltekian, and Grégoire Altan-Bonnet. 2021. Scifive: a text-to-text transformer model for biomedical literature. *CoRR*, abs/2106.03598.

Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, H. Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training gopher. *CoRR*, abs/2112.11446.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Lance Ramshaw and Mitch Marcus. 1995. Text chunking using transformation-based learning. In *Third Workshop on Very Large Corpora*.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Stephen E. Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Alexey Romanov and Chaitanya Shivade. 2018. Lessons from natural language inference in the clinical domain. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1586–1596, Brussels, Belgium. Association for Computational Linguistics.

Devendra Singh Sachan, Siva Reddy, William L. Hamilton, Chris Dyer, and Dani Yogatama. 2021. End-to-end training of multi-document reader and retriever for open-domain question answering. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 25968–25981.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M. Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. 2021. Multitask prompted training enables zero-shot task generalization. *CoRR*, abs/2110.08207.

Minjoon Seo, Tom Kwiatkowski, Ankur Parikh, Ali Farhadi, and Hannaneh Hajishirzi. 2018. Phrase-indexed question answering: A new challenge for scalable document comprehension. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 559–564, Brussels, Belgium. Association for Computational Linguistics.

Minjoon Seo, Jinhyuk Lee, Tom Kwiatkowski, Ankur Parikh, Ali Farhadi, and Hannaneh Hajishirzi. 2019. Real-time open-domain question answering with dense-sparse phrase index. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4430–4441, Florence, Italy. Association for Computational Linguistics.

Hoo-Chang Shin, Yang Zhang, Evelina Bakhturina, Raul Puri, Mostofa Patwary, Mohammad Shoeybi, and Raghav Mani. 2020. BioMegatron: Larger biomedical domain language model. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4700–4706, Online. Association for Computational Linguistics.

Larry Smith, Lorraine K Tanabe, Cheng-Ju Kuo, I Chung, Chun-Nan Hsu, Yu-Shi Lin, Roman Klinger, Christoph M Friedrich, Kuzman Ganchev, Manabu Torii, et al. 2008. Overview of biocreative ii gene mention recognition. *Genome biology*, 9(2):1–19.

Yixuan Su, Zaiqiao Meng, Simon Baker, and Nigel Collier. 2021. Few-shot table-to-text generation with prototype memory. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 910–917, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yi Tay, Mostafa Dehghani, Jinfeng Rao, William Fedus, Samira Abnar, Hyung Won Chung, Sharan Narang, Dani Yogatama, Ashish Vaswani, and Donald Metzler. 2021. Scale efficiently: Insights from pre-training and fine-tuning transformers. *CoRR*, abs/2109.10686.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Robert Tinn, Hao Cheng, Yu Gu, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Fine-tuning large neural language models for biomedical natural language processing. *CoRR*, abs/2112.07869.

Kushal Tirumala, Aram H. Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. 2022. Memorization without overfitting: Analyzing the training dynamics of large language models. *CoRR*, abs/2205.10770.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R. Alvers, Dirk Weissenborn, Anastasia

Krithara, Sergios Petridis, Dimitris Polychronopoulos, Yannis Almirantis, John Pavlopoulos, Nicolas Baskiotis, Patrick Gallinari, Thierry Artières, Axel-Cyrille Ngonga Ngomo, Norman Heino, Éric Gaussier, Liliana Barrio-Alvers, Michael Schroeder, Ion Androutsopoulos, and Georgios Paliouras. 2015. An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. *BMC Bioinform.*, 16:138:1–138:28.

Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerry Tesauro, Bowen Zhou, and Jing Jiang. 2018. R^3: Reinforced ranker-reader for open-domain question answering. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5981–5988. AAAI Press.

Xing Wei and W. Bruce Croft. 2006. Lda-based document models for ad-hoc retrieval. In *SIGIR*, pages 178–185. ACM.

Yuhuai Wu, Markus N. Rabe, DeLesley Hutchins, and Christian Szegedy. 2022. Memorizing transformers. *CoRR*, abs/2203.08913.

Ikuya Yamada, Akari Asai, and Hannaneh Hajishirzi. 2021. Efficient passage retrieval with hashing for open-domain question answering. 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 979–986, Online. Association for Computational Linguistics.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-end open-domain question answering with BERTserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 72–77, Minneapolis, Minnesota. Association for Computational Linguistics.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2022. LinkBERT: Pretraining language models with document links. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8003–8016, Dublin, Ireland. Association for Computational Linguistics.

Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. 2021. QA-GNN: Reasoning with language models and knowledge graphs for question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 535–546, Online. Association for Computational Linguistics.

Dani Yogatama, Cyprien de Masson d'Autume, and Lingpeng Kong. 2021a. Adaptive semiparametric language models. *Transactions of the Association for Computational Linguistics*, 9:362–373.

Dani Yogatama, Cyprien de Masson d'Autume, and Lingpeng Kong. 2021b. Adaptive semiparametric language models. *Transactions of the Association for Computational Linguistics*, 9:362–373.

Hongyi Yuan, Zheng Yuan, Ruyi Gan, Jiaxing Zhang, Yutao Xie, and Sheng Yu. 2022. BioBART: Pretraining and evaluation of a biomedical generative language model. In *Proceedings of the 21st Workshop on Biomedical Language Processing*, pages 97–109, Dublin, Ireland. Association for Computational Linguistics.

Jingyi Zhang, Masao Utiyama, Eiichro Sumita, Graham Neubig, and Satoshi Nakamura. 2018. Guiding neural machine translation with retrieved translation pieces. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1325–1335, New Orleans, Louisiana. Association for Computational Linguistics.

A Biomedical Needs

Compared to the open domain, biomedicine raises substantial challenges and constraints:

- specialized jargon and professional language;
- overarching information truly hard to interpret;
- synonyms (see UMLS) and special tokens;
- narrow margin for interpretation, rephrasing, and creativity;
- clauses are often interdependent and express complex interactions;
- non-tolerance of factual mistakes;
- knowledge rapidly evolves over time.

We cope with these needs by utilizing a domain-specific model, a semantic dense retrieval of commonsense or domain-specific related knowledge (disjoint from the training dataset), and an in-depth evaluation (also with multi-task learning).

B Reproducibility

B.1 Implementation and Training Details

Hardware Setup We ran each experiment on a workstation having one Nvidia GeForce RTX3090 GPU with 24GB of dedicated memory, 64GB of RAM, and an Intel® Core™ i9-10900X1080 CPU @ 3.70GHz.

Model We implement BIOREADER using PyTorch 1.9 (Paszke et al., 2019) as framework, evolving the `T5ForConditionalGeneration` code[5] from HuggingFace and taking LabML[6] and open contributions[7] as references for RETRO-blocks. We only use the T5-base configuration (12 layers, 768-dimensional hidden size, and 12 attention heads) as a baseline due to GPU memory constraints. Nevertheless, we believe that our results would generalize to larger configurations.

Evidence datastore Each abstract in PubMed-RCT gets split into the desired chunk length and then padded if the last chunk is too short. We compute chunk embeddings by taking the mean pooling of the hidden states produced by the encoder[8]. We leverage `Autofaiss`[9] for automatically building the document indices and then calculating the k-nearest neighbors for all chunks. Creating an entire FAISS index on our knowledge base $\mathcal{D}$ with approximately 200K abstracts and 60M tokens takes 2 hours (≈1.5GB index file, ≈0.7GB chunk file). Data leakage is not possible with different sources for queries and neighbors; so we do not filter out neighbors originating from the same document as the training sequence.

Experiment tracking We track all our trainings with Weights & Biases[10] and monitor CO2 emissions with CodeCarbon[11]. Moreover, we profile the neighbors' retrieval speed with custom code.

Pre-training After initializing the model parameters (warm-up) with SCIFIVE(PubMed)-base, we continuously pre-train BIOREADER for 122K steps with a batch size of 8. We take two SCIFIVE pre-training files as our corpus[12]. Here, spans of text (i.e., consecutive tokens) are randomly replaced by a sentinel unique masked token `<M>`; the target sequence consists of the concatenation of the same sentinel tokens and the real dropped-out spans (self-supervised learning). We use Adam (Kingma and Ba, 2015) as optimizer with a constant learning rate of `1e-4` and a dropout rate of 10%. With k=9 and max-length n=512, the training time is ≈10 hours (1 second per iter), 0.2682 kg CO2 impact. We highlight that decreasing the retrieved chunks to k=2 reduces the time required to 0.7 seconds per iter while increasing the max-length to n=1024 leads to >20 hours. We perform the retrieval of all chunks in parallel by putting them into a single batch; retrieving one chunk at a time causes a strong deterioration in performance (≈2.5 days for k=9).

Fine-tuning After pre-training, we fine-tune BIOREADER on the various downstream tasks, choosing a multi-task learning configuration for the NER datasets. Due to its unavailability in the original paper, we re-calculate SCIFIVE Exact Match accuracy for QA. We perform training for 30 epochs with a batch size of 4 for tasks with 256 input length (2 otherwise), AdamW (Loshchilov and Hutter, 2019), learning rate `2e-4`, and dropout rate 10%. We find a large batch size to be very beneficial; we simulate a batch size of 128 with 32 and 64 gradient accumulation steps, thus helping to prevent overfitting. Maximum input and output lengths for each task are in Table 7. Each fine-tuning takes between 13 and 20 hours.

Used models Table 5 enumerates all the models used in this study, linking to specific versions.

B.2 Hyperparameters

We list the hyperparameters used for training BIOREADER in Table 6. An insight into their effect is given in §D.

C Evaluation Datasets Insights

Table 7 reports a complete overview of our benchmark datasets and their composition.

D Ablations

We study several research questions to understand the effect of important design choices and hyperparameters on downstream biomedical NLP performance. We test on lightweight settings to save computation time without affecting comparability. We pre-train on ≈15K instances[13] and fine-tune for

[5] https://huggingface.co/docs/transformers/model_doc/t5

[6] https://nn.labml.ai/transformers/retro/model.html

[7] https://github.com/lucidrains/RETRO-pytorch

[8] We also tried `[CLS]` but found no consistent best strategy (the optimal one varies on different encoders).

[9] https://github.com/criteo/autofaiss

[10] https://wandb.ai

[11] https://github.com/mlco2/codecarbon

[12] Masked pre-training files: gs://scifive/pretrain/pubmed_cleaned/abs_1_30.tsv and gs://scifive/pretrain/pubmed_cleaned/abs_1_16.tsv

[13] gs://scifive/pretrain/pubmed_cleaned/abs_1_23.tsv

Model	URL
BioBERT	https://huggingface.co/dmis-lab/biobert-base-cased-v1.1
BioBERT-NLI	https://huggingface.co/gsarti/biobert-nli
SciBERT	https://huggingface.co/allenai/scibert_scivocab_cased
BlueBERT	https://huggingface.co/bionlp/bluebert_pubmed_mimic_uncased_L-12_H-768_A-12
ClinicalBERT	https://huggingface.co/emilyalsentzer/Bio_ClinicalBERT
PubMedBERT-base	https://huggingface.co/microsoft/BiomedNLP-PubMedBERT-base-uncased-abstract
BioLinkBERT-base	https://huggingface.co/michiyasunaga/BioLinkBERT-base
BioLinkBERT-large	https://huggingface.co/michiyasunaga/BioLinkBERT-large
T5-base	https://huggingface.co/t5-base
T5-large	https://huggingface.co/t5-large
SciFive-base	https://huggingface.co/razent/SciFive-base-Pubmed
SciFive-large	https://huggingface.co/razent/SciFive-large-Pubmed_PMC

Table 5: List of the models used in this study.

Hyperparameter	Search space
Pre-training learning rate	{1e−5, 3e−5, 5e−5, 1e−4∗ (Yuan et al., 2022), 2e−4 (Borgeaud et al., 2021)}
Fine-tuning learning rate	2e−4 (linear scheduler)
Pre-training and fine-tuning dropout rate	0.10
Pre-training Optimizer	Adam
Fine-tuning Optimizer	AdamW, (0.9 β_1, 0.99 β_2, no weight decay)
Pre-training batch size	{2, 6, 8∗}
Fine-tuning batch size	4 for RE, DC, and NLI, 2 for NER, QA, and OpenQA, gradient_accumulation_steps=32
Pre-training iterations on PubMed sample	122K (0.3 epochs)
Fine-tuning epochs on downstream tasks	30
Pre-training and fine-tuning CCA position P	{6, 9, 12}, {9, 12}∗, {6, 9}, {12}
Pre-training and fine-tuning number of neighbors k	[2, 9] (9∗)
Chunk size m	{8,16∗, 32 (Borgeaud et al., 2021)}
Pre-training and fine-tuning checkpoint frequency	10.000 steps

Table 6: Hyperparameters along with their search grid. ∗ marks the values used to obtain the reported results.

one epoch on a small corpus version of each downstream task (5k train data and 500 test data), batch size of 2. We exclude QA and OpenQA benchmarks due to their need for human judgment as a proper quality indication (§5).

RQ1. What is the best architectural setting (position and quantity) for the CCA layers? We study three architecture variants for CCA layers.

- *Encoder-only (Enc).* CCA is done within the encoder after the standard self-attention layer.
- *Encoder-Decoder (EncDec).* CCA is done between the encoder and the decoder. After the encoder, the encoded retrieved neighbors are incorporated once with the encoder output through CCA and are saved as an independent variable. In the decoder, there is a second layer of Encoder-Decoder Attention to blend the CCA output with the decoder hidden states.
- *Decoder-only (Dec).* CCA is done within the decoder, after the standard self-attention layer, and before the Encoder-Decoder Attention layer. Encoded neighbors are integrated into encoder outputs with CCA and replace the encoder outputs themselves.

We find that the *Dec* architecture is the best setting for the CCA layers, while the *EncDec* architecture is a close second. The *Enc* architecture is not effective, and we hypothesize that it is important for the raw inputs without neighbor information to be initially seen by the decoder. Furthermore, we find that 2 layers for CCA in a 12-layer model represent an optimal setting. The best results are obtained by comparing in contrast to 3 and 4 CCA layers.

RQ2. How does the chunk size impact the results? We examine how the chunk size (m in Eq.

Task	Dataset	Biomedical Domain	# Instances			Task Type	Input Length	Target Length
			Train	Dev	Test			
Self-Supervised Learning	PubMed	All					512	512
NER	NCBI-disease	Disease	5,134	787	960	Multi-Task	512	512
	BC5CDR-disease	Disease	4,182	4,244	4,424			
	BC5CDR-chemical	Chemical	5,203	5,347	5,538			
	BC4CHEMD	Chemical	30,682	26,364	26,364			
	BC2GM	Chemical	12,574	5,038	5,038			
	JNLPBA	Gene	46,750	4,551	8,662			
	Species-800	Species	10,771	1,630	1,630			
RE	Chemprot	Protein-chemical	18,035	11,268	15,745	Single-Task	256	16
	DDI	Disease-disease	25,296	2,496	5,716	Single-Task	256	16
DC	HoC	Cancer	1,295	186	371	Single-Task	256	64
NLI	MedNLI	Clinical	11,232	1,395	1,422	Single-Task	256	12
QA	BioASQ4-factoid	All	3,264	3,590	652	Single-Task	512	128
	BioASQ5-factoid	All	3,264	496	495	Single-Task	512	128
	BioASQ6-factoid	All	4,772	478	531	Single-Task	512	128
OpenQA	MedQA-USMLE	Clinical	10,178	1,272	1,273	Single-Task	512	128

Table 7: Basic statistics of the biomedical evaluation datasets with input and target sequence length settings, including self-supervised learning. "# Instances" denotes the number of entity or relation mentions (NER, RE) / labeled documents or sentence pairs (DC, NLI) / queries (QA, OpenQA).

5) affects the model performance. To this end, we re-build the datastore $\mathcal{D}$ and re-run pre-training and fine-tuning by varying $m \in \{8, 16, 32\}$, where 32 represents the same input to chunk ratio as RETRO. Results are quite similar (Figure 3). Surprisingly, m=16 has average better accuracies/F1-scores. After scanning some neighborhood examples, we believe this is due to the greater compression of information content within RCTs' abstracts. Reasonably, the increase in the chunk size is directly proportional to the memory occupation and inversely proportional to the computation time required (23 minutes for m=8, 12 minutes for m=16, 7 minutes for m=32). Low NER F1 scores are justified by the need for more training data and time to accomplish adaption. Clearly, the text-to-text instances belonging to this task type are more distant from human language due to entity labels directly inserted in the text through augmentation.

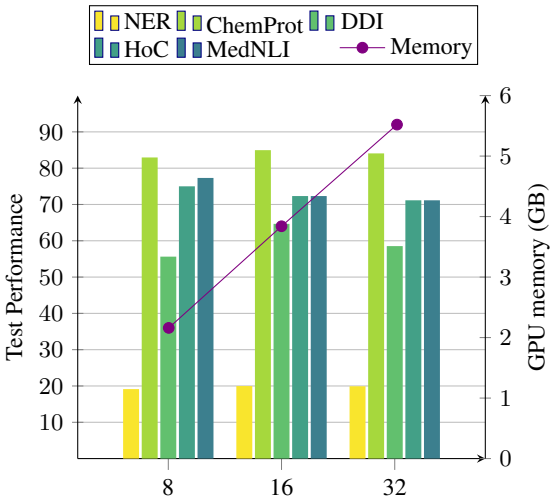

Figure 3: Overall test performance and GPU memory occupation for different chunk sizes (*Dec* CCA, SCIFIVE neighbor encoder).

RQ3. What is the most effective neural retriever for building the evidence datastore? The provenance of the continuous representations used for the neural retrieval phase is pivotal. Previous work like RETRO exploits BERT embeddings independently of the architecture, assuming that the non-frozen encoder part of the model will learn to adapt. We explore different frozen bi-directional models for encoding neighbors within the datastore $\mathcal{D}$: (i) PUBMEDBERT—one of the most effective biomed-

ical *BERT (Liu et al., 2021), (ii) BIOBERT-NLI sentence transformer—pre-trained on sentence similarity, (iii) the SCIFIVE encoder, and (iv) the CONTRIEVER query/document encoder. Table 8 shows the results. We can see that utilizing a CONTRIEVER-based encoder for both chunked input prompts (queries) and neighbors give general better results, rewarding space-homogeneity. Moreover, from qualitative analysis, we find that the tokens retrieved by CONTRIEVER are more relevant than the ones obtained through the SCIFIVE-encoder.

Dataset (Metric)	PubMedBert	BioBert	SciFive	Contriever
ChemProt (F1)	81.82	83.84	87.60	**87.77**
DDI (F1)	51.65	60.32	**64.70**	49.91
MedNLI (Acc)	63.36	72.86	**76.71**	75.59
NER (F1)	15.17	16.82	18.31	**18.33**
HoC (F1*)	60.28	57.84	64.70	**77.06**

Table 8: Downstream test results with different query-neighbors encoders (chunk size 16, *Dec* CCA). Best scores are in bold.

RQ4. What is the contribution of continuation chunks? A BIOREADER model is trained by attending, for a given chunk, to both the neighbors of the preceding chunk N and their continuation F in time. We measure how training and evaluating only on neighbors affects performance (Table 9). We observe that attending to both neighbors and their continuation is generally the most effective choice. One exception to this claim is NLI, for which we register a decrease in accuracy of more than 20 points. We believe it is normal behavior: as the external context increases (i.e., higher k-values or continuations), the model tends to divert attention from the two sentences under evaluation and make erroneous predictions.

Dataset (Metric)	*N*-only	*N* + *F*
NER (F1)	19.37	**19.98**
ChemProt (F1)	80.79	**84.96**
DDI (F1)	58.79	**64.66**
HoC (F1*)	70.42	**72.32**
MedNLI (Acc)	**76.54**	52.63

Table 9: Downstream test results with and without continuation chunks (chunk size 16, *Dec* CCA, SCIFIVE neighbor encoder). Best scores are in bold.

RQ5. What is the impact of the number of training neighbors? During training, we retrieve the top-k neighboring chunks for each query. We weigh the effect of training with multiple numbers of neighbors, considering $k \in \{2, 3, \ldots, 9\}$. Figure 4 summarizes the resulting performance. We find that results are quite stable within the small tested range, with no particular k-value giving substantial performance improvement. We emphasize that a simple solution for reducing the required computational budget consists in training the model with fewer retrieval passages. In this paper, we select k=9 due to the tiny superior performance and the contained overhead (see §B). We have the flexibility to adjust the number of retrieved neighbors at evaluation time, which can affect performance and runtime.

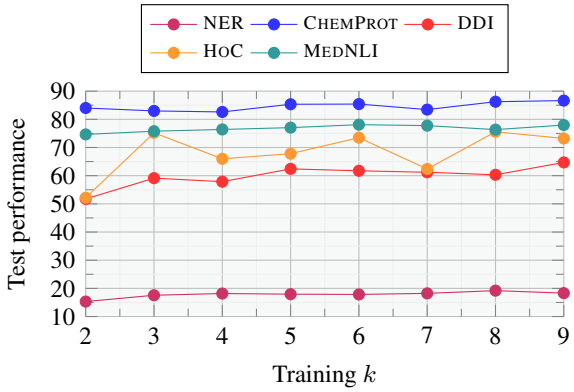

Figure 4: Impact of the number of nearest neighbors k during training (chunk size 16, *Dec* CCA, SCIFIVE neighbor encoder).

RQ6. How the model scales with the number of retrieved passages during the evaluation? We investigate the performance of BIOREADER as we vary k during evaluation. In a general way, we observe that when k is small ($2<k<15$), performances are relatively the same. However, as k approaches 30, the results drop notably (more than 2 points less on average). The reason for such degeneration is that, as k increases, the top-k neighbors are likely to contain more information that is irrelevant to the input prompt or repeated by other chunks. k=1 or k=2 lead to minor improvements in non-knowledge-intensive tasks like RE or NLI.

E Loss and Perplexity

We outline the loss and perplexity curves at the end of the pre-training process (Figure 5). In doing this, we compare BIOREADER with our baseline, i.e., training continuation of SCIFIVE-base with all the layers unfrozen (no architectural changes, no neighbors, no CCA). Both the loss for tokens and the perplexity (which indicates better generaliza-

tion performance) are reduced by BIOREADER in a pronounced way.

F Visualization

F.1 Chunked-Cross Attention

Figure 6 illustrates the simplified functioning of chunked-cross attention, the step where the model can glance at the external information it needs to correctly predict the next token.

F.2 Input-Neighbors-Output Examples

We check out how the retrieved chunks guide the decoding (Table 10), seeking overlapping between sampled and neighbor tokens. Using a contextual-aware PLM encoder, we capture lexical variations and semantic relationships between the input prompt and the searched chunks. Retrieval supplies more insights on the output of BIOREADER, as the user can directly visualize or modify the neighbors that are being used. One can also verify the source documents from which the utilized knowledge originates, which means our model also has increased interpretability and debuggability compared to standard language models. So, BIOREADER engenders appropriate user trust, supporting a great understanding of the modeled process. We find that BIOREADER uses its non-parametric memory to cue the encoder-decoder model into generating correct tokens. We note that the role of the retrieved knowledge changes depending on the task. In non-knowledge intensive scenarios, neighbors offer an extended view of the meaning of the phrases mentioned in the input prompt, giving to BIOREADER related examples that can be helpful to predict a class label better. In knowledge-intensive tasks requiring free-text generation, retrieved neighbors suggest factual evidence fragments that guide the construction of the output text token-after-token, reducing hallucinations and making the model more knowledgeable.

F.3 Zero-shot Generalization Via Datastore Update

Figure 7 displays a graphical representation of the BIOREADER output in Table 3. To the best of our knowledge, we are the first to test PLM transfer learning by explicit memory substitution only instead of closed-book generalization. For this reason, we coin the term *"zero-shot datastore"*.

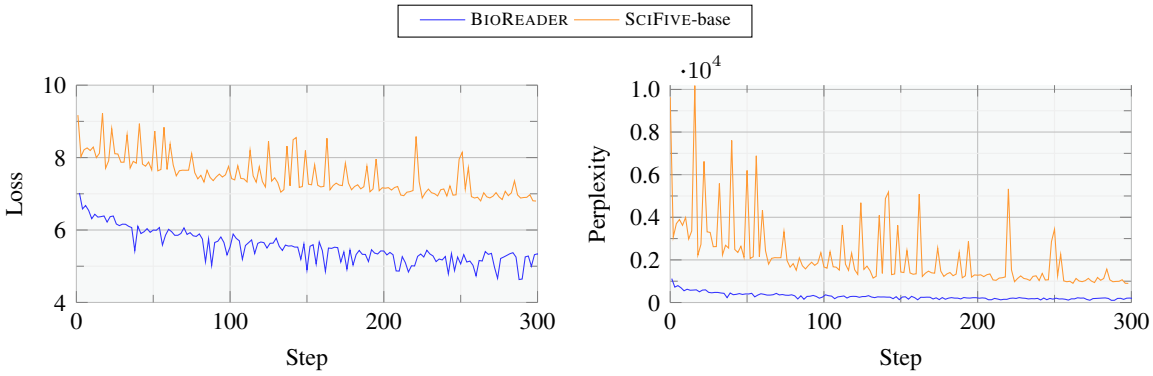

Figure 5: Loss and perplexity curves after pre-training.

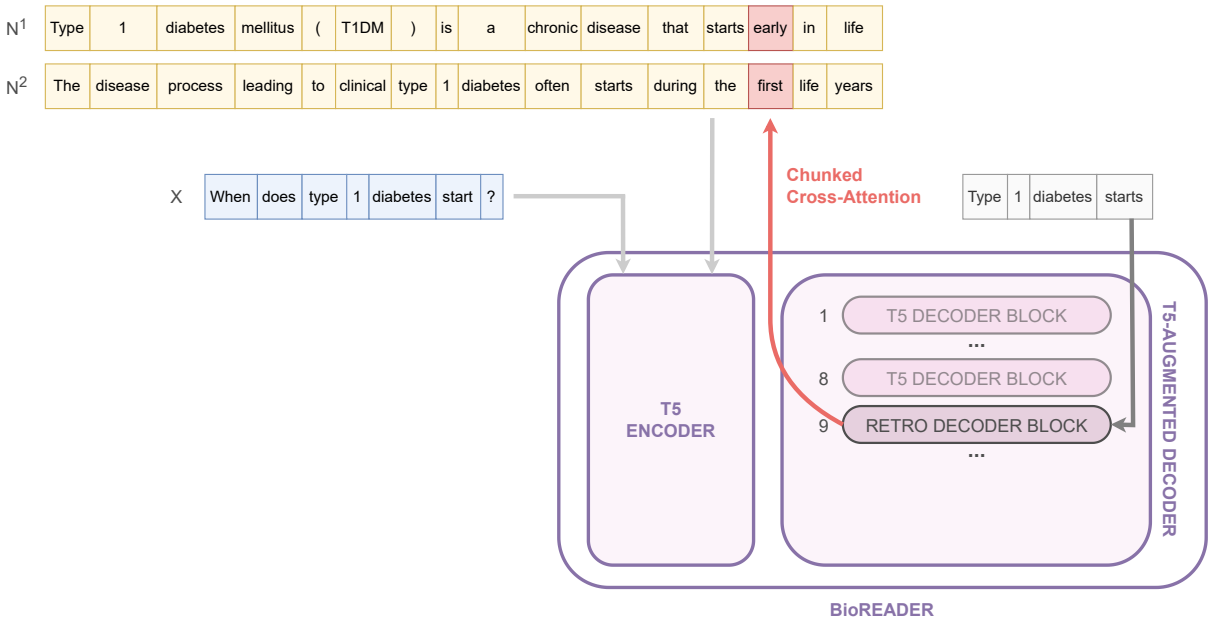

Figure 6: BioReader decoder block retrieving information from nearest neighbor chunks using Cca. Adapted from https://jalammar.github.io/illustrated-retrieval-transformer/.

Task	Input	Output	N_1^{1-3}	N_2^{1-3}	N_3^{1-3}
RE	ddi: The concomitant intake of * DRUG * and * DRUG * does not affect the pharmacokinetics of either alcohol or acamprosate.	*Ground truth:* DDI-false *Ours:* DDI-false	This was a multicentre, randomised, double-blind, placebo - and active-controlled	require more medication than younger children to achieve a similar therapeutic response	greater ease of administration when compared with oxytocin
			used in these doses seems to be safe for day care surgery	with only a tiny increase in circulating plasma.	the uniform injection of vaccine antigen into muscle tissue in infants.
			No adverse events could be related to the use PSD .	Habitual caffeine use appears to minimally reduce caffeine effects .	to sodium benzoate containing pharmaceutical formulations
DC	hoc: The present study suggests that MGN-3 may represent an immunologically relevant product for activating innate immunity in multiple myeloma patients and warrants further testing to demonstrate clinical efficacy	*Ground truth:* avoiding immune destruction *Ours:* avoiding immune destruction	MGN-3/BioBran is an arabinoxylan extracted from rise bran	cell-mediated immune response plays a role in wart resolution.	from their disease and therapy .
			inflammatory response through modulation of the neurohumoral response to stress.	efficacy by way of modulating cellular immune function .	a probe for challenge studies .
			boosting the immune system .	decrease in receptor-mediated apoptosis .	can be a diagnostic parameter .
QA	bioasq4b: question*: what is targeted by monoclonal antibody pembrolizumab ? context*: pembrolizumab versus ipilimumab in advanced melanoma . background: the immune checkpoint inhibitor ipilimumab is the standard-of-care treatment for patients with advanced melanoma. pembrolizumab inhibits the programmed cell death 1 (pd-1) immune checkpoint and has antitumor activity in patients with advanced melanoma . . .	*Ground truth:* programmed cell death 1 *Ours:* pd-1	bevacizumab when added to standard chemotherapy in a real-world	In randomized sequence, patients received oral montelukast	Rotavirus is a leading cause of morbidity and mortality in children younger
			a novel selective estrogen-receptor modulator, in postmeno	lerability and safety of TMC278, a non-nu	Ziprasidone is not currently approved by the United States Food and Drug Administration
			acy and safety of Fibrocaps, a ready-to-use	acy and safety of Fibrocaps, a ready-to-use	Although platin-based chemotherapy has become a standard treatment for non-small cell
OpenQA	medqa: question*: A 73-year-old man has type 2 diabetes mellitus, hypertension , hypercholesterolemia , and coronary artery disease . The physician prescribes a drug that inhibits intestinal cholesterol absorption. The addition of this drug is most likely to increase the risk of which of the following adverse effects? Hepatotoxicity Hyperkalemia Cutaneous flushing Hyperuricemia	*Ground truth:* Hepatotoxicity *Ours:* Hepatotoxicity	Arterial hypertension is a prime cause of morbidity and mortality in	years or younger who were discharged from the hospital after a coronary heart disease	hypothalamic cholinergic neurotransmission plays a major
			Obesity is a highly prevalent medical condition and is commonly accompanied by	ised on a cholesterol-lowering diet and simvastatin 40 mg daily	hepatotoxicity have been observed in obese patients
			Capillary glucose levels decreased by 2.9 and 2.6 mmol	obesity-related renal failure after lower torso ischemia	hyperglycemia and the frequency of white!75!greenhypoglycemia. we conducted

Table 10: Cherry picked input-output examples and retrieval influence. We show the first three neighbors N_u^{1-3} of the chunks $u \in \{1, 2, 3\}$. We highlight the latent semantic overlap between the input and the retrieved neighbors.

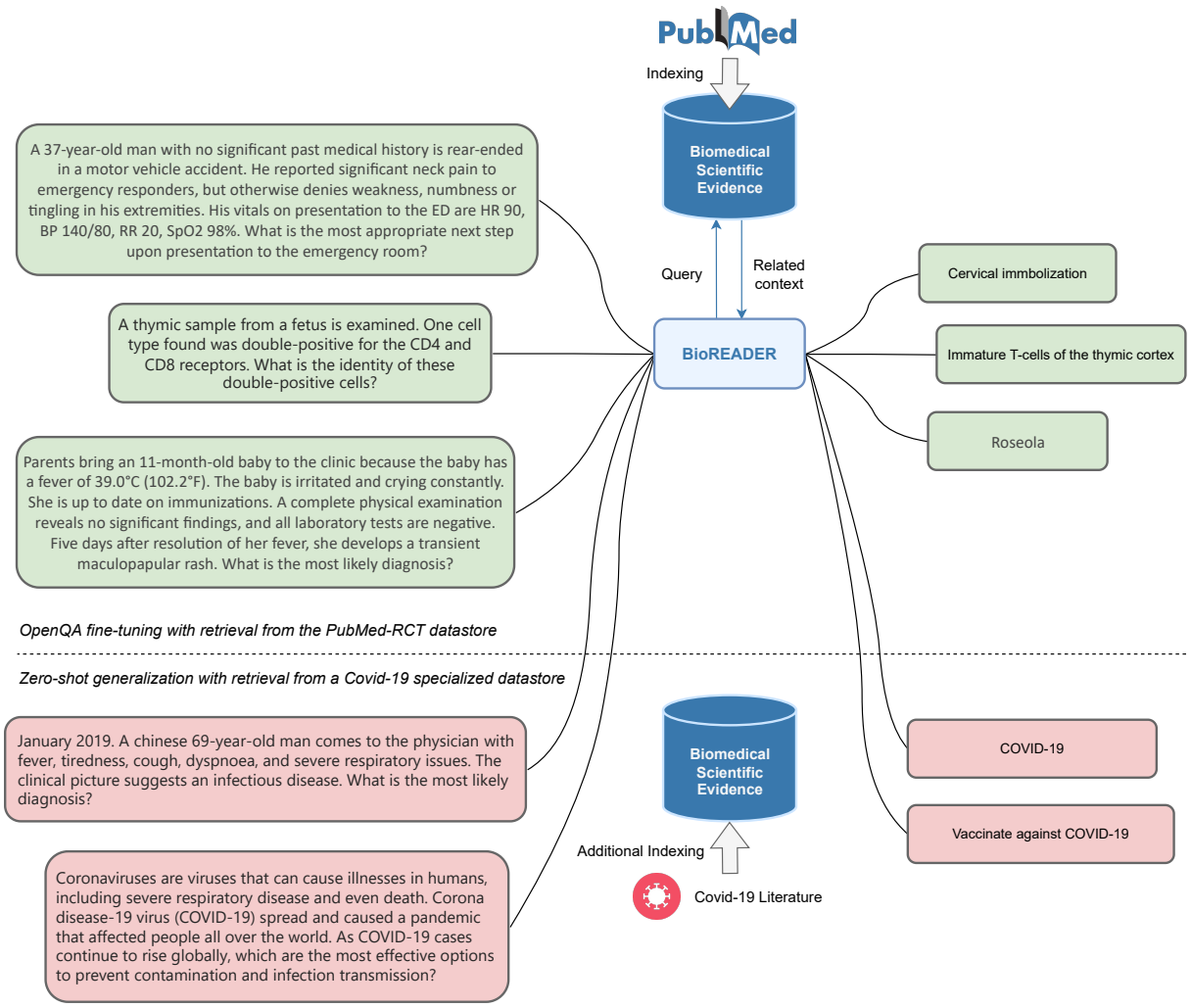

Figure 7: BIOREADER adapts and provides correct answers to unseen context-free Covid-19 questions only through a datastore enrichment (no retraining).