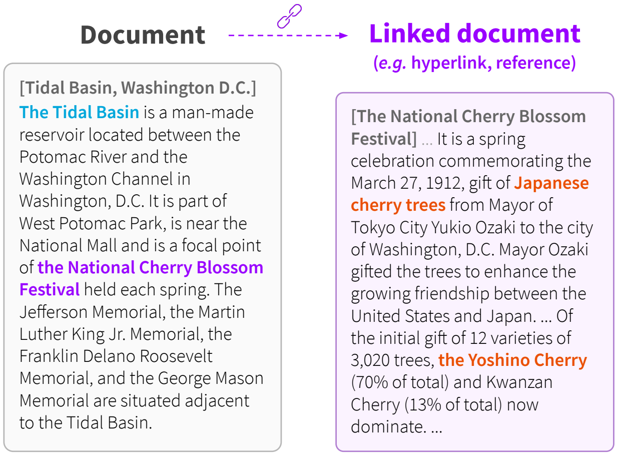
Link-BERT: Pretraining a Language Model with Document Links

Michihiro Yasunaga,[1] **Jure Leskovec,**[1*] **Percy Liang**[1*]
[1]Stanford University *Equal senior authorship
{myasu,jure,pliang}@cs.stanford.edu

Abstract

Language model (LM) pretraining can learn various knowledge from text corpora, helping downstream tasks. However, existing methods such as BERT model a single document, and do not capture dependencies or knowledge that span across documents. In this work, we propose *LinkBERT*, an LM pretraining method that leverages links between documents, e.g., hyperlinks. Given a text corpus, we view it as a graph of documents and create LM inputs by placing linked documents in the same context. We then pretrain the LM with two joint self-supervised objectives: masked language modeling and our new proposal, document relation prediction. We show that LinkBERT outperforms BERT on various downstream tasks across two domains: the general domain (pretrained on Wikipedia with hyperlinks) and biomedical domain (pretrained on PubMed with citation links). LinkBERT is especially effective for multi-hop reasoning and few-shot QA (+5% absolute improvement on HotpotQA and TriviaQA), and our biomedical LinkBERT sets new states of the art on various BioNLP tasks (+7% on BioASQ and USMLE). We will release our pretrained models, *LinkBERT* and *BioLinkBERT*, as well as code and data at https://github.com/michiyasunaga/LinkBERT.



[Tidal Basin, Washington D.C.]
The Tidal Basin is a man-made reservoir located between the Potomac River and the Washington Channel in Washington, D.C. It is part of West Potomac Park, is near the National Mall and is a focal point of **the National Cherry Blossom Festival** held each spring. The Jefferson Memorial, the Martin Luther King Jr. Memorial, the Franklin Delano Roosevelt Memorial, and the George Mason Memorial are situated adjacent to the Tidal Basin.

[The National Cherry Blossom Festival] ... It is a spring celebration commemorating the March 27, 1912, gift of **Japanese cherry trees** from Mayor of Tokyo City Yukio Ozaki to the city of Washington, D.C. Mayor Ozaki gifted the trees to enhance the growing friendship between the United States and Japan. ... Of the initial gift of 12 varieties of 3,020 trees, **the Yoshino Cherry** (70% of total) and Kwanzan Cherry (13% of total) now dominate. ...

Figure 1: Document links (e.g. hyperlinks) can provide salient multi-hop knowledge. For instance, the Wikipedia article "Tidal Basin" (left) describes that the basin hosts "National Cherry Blossom Festival". The hyperlinked article (right) reveals that the festival celebrates "Japanese cherry trees". Taken together, the link suggests new knowledge not available in a single document (e.g. "Tidal Basin has Japanese cherry trees"), which can be useful for various applications, including answering a question "What trees can you see at Tidal Basin?". We aim to leverage document links to incorporate more knowledge into language model pretraining.

1 Introduction

Pretrained language models (LMs), like BERT and GPTs (Devlin et al. 2019; Brown et al. 2020), have shown remarkable performance on many natural language processing (NLP) tasks, such as text classification and question answering, becoming the foundation of modern NLP systems (Bommasani et al. 2021). By performing self-supervised learning, such as masked language modeling (Devlin et al. 2019), LMs learn to encode various knowledge from text corpora and produce informative representations for downstream tasks (Petroni et al. 2019; Bosselut et al. 2019; Raffel et al. 2020).

However, existing LM pretraining methods typically consider text from a single document in each input context (Liu et al. 2019; Joshi et al. 2020) and do not model links between documents. This can pose limitations because documents often have rich dependencies (e.g. hyperlinks, references), and knowledge can span *across* documents. As an example, in Figure 1, the Wikipedia article "Tidal Basin, Washington D.C." (left) describes that the basin hosts "National Cherry Blossom Festival", and the hyperlinked article (right) reveals the background that the festival celebrates "Japanese cherry trees". Taken together, the hyperlink offers new, multi-hop knowledge "Tidal Basin has Japanese cherry trees", which is not available in the single article "Tidal Basin" alone. Acquiring such multi-hop knowledge in pretraining could be useful for various applications including question answering. In fact, document links like hyperlinks and references are ubiquitous (e.g. web, books, scientific literature), and guide how we humans acquire knowledge and even make discoveries (Margolis et al. 1999).

In this work, we propose *LinkBERT*, an effective language model pretraining method that incorporates document link knowledge. Given a text corpus, we obtain links between documents such as hyperlinks, and create LM inputs by placing linked documents in the same context, besides the existing option of placing a single document or random documents as in BERT. Specifically, as in Figure 2, after sampling an anchor text segment, we place either (1) the contiguous segment from the same document, (2) a random document, or (3) a document linked from anchor segment, as the next segment in the input. We then train the LM with two joint objectives: We use masked language modeling (MLM) to encourage learning multi-hop knowledge of concepts brought into the same context by

Accepted to Workshop on Knowledge Augmented Methods for Natural Language Processing, in conjunction with AAAI 2023.

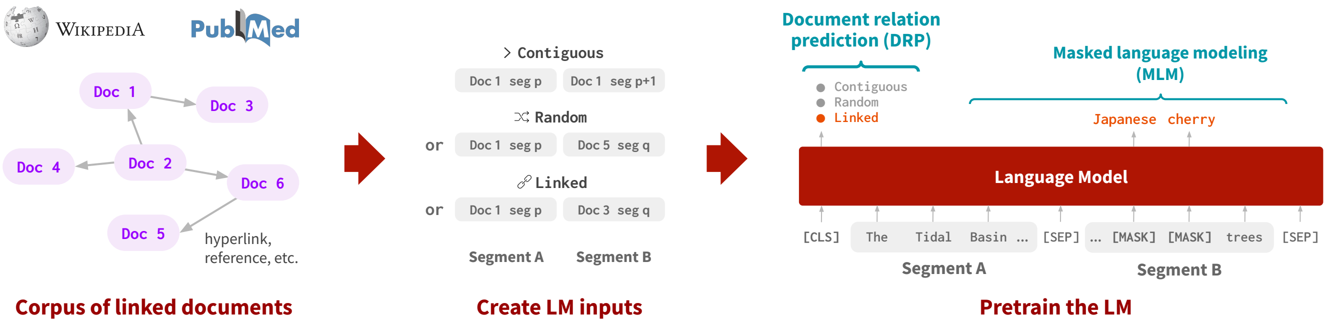


Figure 2: **Overview of our approach, LinkBERT**. Given a pretraining corpus, we view it as a graph of documents, with links such as hyperlinks (§4.1). To incorporate the document link knowledge into LM pretraining, we create LM inputs by placing a pair of linked documents in the same context (*linked*), besides the existing options of placing a single document (*contiguous*) or a pair of random documents (*random*) as in BERT. We then train the LM with two self-supervised objectives: masked language modeling (MLM), which predicts masked tokens in the input, and document relation prediction (DRP), which classifies the relation of the two text segments in the input (*contiguous*, *random*, or *linked*) (§4.2).

document links (e.g. "Tidal Basin" and "Japanese cherry" in Figure 1). Simultaneously, we propose a Document Relation Prediction (DRP) objective, which classifies the relation of the second segment to the first segment (*contiguous*, *random*, or *linked*). DRP encourages learning the relevance and bridging concepts (e.g. "National Cherry Blossom Festival") between documents, beyond the ability learned in the vanilla next sentence prediction objective in BERT.

Viewing the pretraining corpus as a graph of documents, LinkBERT is also motivated as self-supervised learning on the graph, where DRP and MLM correspond to link prediction and node feature prediction in graph machine learning (Yang et al. 2015; Hu et al. 2020). Our modeling approach thus provides a natural fusion of language-based and graph-based self-supervised learning.

We train LinkBERT in two domains: the general domain, using Wikipedia articles with hyperlinks (§4), and the biomedical domain, using PubMed articles with citation links (§6). We then evaluate the pretrained models on a wide range of downstream tasks such as question answering, in both domains. LinkBERT consistently improves on baseline LMs across domains and tasks. For the general domain, LinkBERT outperforms BERT on MRQA benchmark (+4% absolute in F1-score) as well as GLUE benchmark. For the biomedical domain, LinkBERT exceeds PubmedBERT (Gu et al. 2020) and sets new states of the art on BLURB biomedical NLP benchmark (+3% absolute in BLURB score) and MedQA-USMLE reasoning task (+7% absolute in accuracy). Overall, LinkBERT attains notably large gains for multi-hop reasoning, multi-document understanding, and few-shot question answering, suggesting that LinkBERT internalizes significantly more knowledge than existing LMs by pretraining with document link information.

2 Related work

Retrieval-augmented LMs. Several works (Lewis et al. 2020b; Karpukhin et al. 2020; Oguz et al. 2020; Xie et al. 2022) introduce a retrieval module for LMs, where given an anchor text (e.g. question), retrieved text is added to the same LM context to improve model inference (e.g. answer prediction). These works show the promise of placing related documents in the same LM context at inference time, but they do not study the effect of doing so in pretraining. Guu et al. (2020) pretrain an LM with a retriever that learns to retrieve text for answering masked tokens in the anchor text. In contrast, our focus is not on retrieval, but on pretraining a general-purpose LM that *internalizes* knowledge that spans across documents, which is orthogonal to the above works (e.g., our pretrained LM could be used to initialize the LM component of these works). Additionally, we focus on incorporating document links such as hyperlinks, which can offer salient knowledge that common lexical retrieval methods may not provide (Asai et al. 2020).

Pretrain LMs with related documents. Several concurrent works use multiple related documents to pretrain LMs. Caciularu et al. (2021) place documents (news articles) about the same topic into the same LM context, and Levine et al. (2021) place sentences of high lexical similarity into the same context. Our work provides a general method to incorporate document links into LM pretraining, where lexical or topical similarity can be one instance of document links, besides hyperlinks. We focus on hyperlinks in this work, because we find they can bring in salient knowledge that may not be obvious via lexical similarity, and yield a more performant LM (§5.5). Additionally, we propose the DRP objective, which improves modeling multiple documents and relations between them in LMs (§5.5).

Hyperlinks and citation links for NLP. Hyperlinks are often used to learn better retrieval models. Chang et al. (2020); Asai et al. (2020); Seonwoo et al. (2021) use Wikipedia hyperlinks to train retrievers for open-domain question answering. Ma et al. (2021) study various hyperlink-aware pretraining tasks for retrieval. While these works use hyperlinks to learn retrievers, we focus on using hyperlinks to create better context for learning general-purpose LMs. Separately, Calixto et al. (2021) use Wikipedia hyperlinks to learn multilingual LMs. Citation links are often used to improve summarization and recommendation of academic papers (Qazvinian & Radev 2008; Yasunaga et al. 2019; Bhagavatula et al. 2018; Khadka et al. 2020; Cohan et al. 2020). Here we leverage citation networks to improve pretraining general-purpose LMs.

Graph-augmented LMs. Several works augment LMs with graphs, typically, knowledge graphs (KGs) where the nodes capture entities and edges their relations. Zhang et al. (2019); He et al. (2020); Wang et al. (2021b) combine LM training with KG embeddings. Sun et al. (2020); Yasunaga et al. (2021); Zhang et al. (2022) combine LMs and graph neural networks (GNNs) to jointly train on text and KGs. Different from KGs, we use document graphs to learn knowledge that spans across documents.

3 Preliminaries

A language model (LM) can be pretrained from a corpus of documents, $\mathcal{X} = \{X^{(i)}\}$. An LM is a composition of two functions, $f_{\text{head}}(f_{\text{enc}}(X))$, where the encoder f_{enc} takes in a sequence of tokens $X = (x_1, x_2, ..., x_n)$ and produces a contextualized vector representation for each token, $(\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_n)$. The head f_{head} uses these representations to perform self-supervised tasks in the pretraining step and to perform downstream tasks in the fine-tuning step. We build on BERT (Devlin et al. 2019), which pretrains an LM with the following two self-supervised tasks.

Masked language modeling (MLM). Given a sequence of tokens X, a subset of tokens $Y \subseteq X$ is masked, and the task is to predict the original tokens from the modified input. Y accounts for 15% of the tokens in X; of those, 80% are replaced with `[MASK]`, 10% with a random token, and 10% are kept unchanged.

Next sentence prediction (NSP). The NSP task takes two text segments (X_A, X_B) as input, and predicts whether X_B is the direct continuation of X_A. Specifically, BERT first samples X_A from the corpus, and then either (1) takes the next segment X_B from the same document, or (2) samples X_B from a random document in the corpus. The two segments are joined via special tokens to form an input instance, `[CLS]` X_A `[SEP]` X_B `[SEP]`, where the prediction target of `[CLS]` is whether X_B indeed follows X_A (*contiguous* or *random*).

In this work, we will further incorporate document link information into LM pretraining. Our approach (§4) will build on MLM and NSP.

4 LinkBERT

We present LinkBERT, a self-supervised pretraining approach that aims to internalize more knowledge into LMs using document link information. Specifically, as shown in Figure 2, instead of viewing the pretraining corpus as a set of documents $\mathcal{X} = \{X^{(i)}\}$, we view it as a *graph* of documents, $\mathcal{G} = (\mathcal{X}, \mathcal{E})$, where $\mathcal{E} = \{(X^{(i)}, X^{(j)})\}$ denotes links between documents (§4.1). The links can be existing hyperlinks, or could be built by other methods that capture document relevance. We then consider pretraining tasks for learning from document links (§4.2): We create LM inputs by placing linked documents in the same context window, besides the existing options of a single document or random documents. We use the MLM task to learn concepts brought together in the context by document links, and we also introduce the Document Relation Prediction (DRP) task to learn relations between documents. Finally, we discuss strategies for obtaining informative pairs of linked documents to feed into LM pretraining (§4.3).

4.1 Document graph

Given a pretraining corpus, we link related documents so that the links can bring together knowledge that is not available in single documents. We focus on hyperlinks, e.g., hyperlinks of Wikipedia articles (§5) and citation links of academic articles (§6). Hyperlinks have a number of advantages. They provide background knowledge about concepts that the document writers deemed useful—the links are likely to have high precision of relevance, and can also bring in relevant documents that may not be obvious via lexical similarity alone (e.g., in Figure 1, while the hyperlinked article mentions "Japanese" and "Yoshino" cherry trees, these words do not appear in the anchor article). Hyperlinks are also ubiquitous on the web and easily gathered at scale (Aghajanyan et al. 2021). To construct the document graph, we simply make a directed edge $(X^{(i)}, X^{(j)})$ if there is a hyperlink from document $X^{(i)}$ to document $X^{(j)}$.

For comparison, we also experiment with a document graph built by lexical similarity between documents. For each document $X^{(i)}$, we use the common TF-IDF cosine similarity metric (Chen et al. 2017; Yasunaga et al. 2017) to obtain top-k documents $X^{(j)}$'s and make edges $(X^{(i)}, X^{(j)})$. We use $k = 5$.

4.2 Pretraining tasks

Creating input instances. Several works (Gao et al. 2021; Levine et al. 2021) find that LMs can learn stronger dependencies between words that were shown together in the same context during training, than words that were not. To effectively learn knowledge that spans across documents, we create LM inputs by placing linked documents in the same context window, besides the existing option of a single document or random documents. Specifically, we first sample an anchor text segment from the corpus (Segment A; $X_A \subseteq X^{(i)}$). For the next segment (Segment B; X_B), we either (1) use the contiguous segment from the same document ($X_B \subseteq X^{(i)}$), (2) sample a segment from a random document ($X_B \subseteq X^{(j)}$ where $j \neq i$), or (3) sample a segment from one of the documents linked from Segment A ($X_B \subseteq X^{(j)}$ where $(X^{(i)}, X^{(j)}) \in \mathcal{E}$). We then join the two segments via special tokens to form an input instance: `[CLS]` X_A `[SEP]` X_B `[SEP]`.

Training objectives. To train the LM, we use two objectives. The first is the MLM objective to encourage the LM to learn multi-hop knowledge of concepts brought into the same context by document links. The second objective, which we propose, is Document Relation Prediction (DRP), which classifies the relation r of segment X_B to segment X_A ($r \in \{\textit{contiguous}, \textit{random}, \textit{linked}\}$). By distinguishing *linked* from *contiguous* and *random*, DRP encourages the LM to learn the relevance and existence of bridging concepts between documents, besides the capability learned in the vanilla NSP objective. To predict r, we use the representation of `[CLS]` token, as in NSP. Taken together, we optimize:

$$\mathcal{L} = \mathcal{L}_{\text{MLM}} + \mathcal{L}_{\text{DRP}} \tag{1}$$

$$= -\sum_i \log\ p(x_i \mid \mathbf{h}_i) - \log\ p(r \mid \mathbf{h}_{\texttt{[CLS]}}) \tag{2}$$

where x_i is each token of the input instance, `[CLS]` X_A `[SEP]` X_B `[SEP]`, and $\mathbf{h}_i$ is its representation.

Graph machine learning perspective. Our two pretraining tasks, MLM and DRP, are also motivated as graph self-supervised learning on the document graph. In graph self-supervised learning, two types of tasks, node feature prediction and link prediction, are commonly used to learn the content and structure of a graph. In node feature prediction (Hu et al. 2020), some features of a node are masked, and the task is to predict them using neighbor nodes. This corresponds to our MLM task, where masked tokens in Segment A can be predicted using Segment B (a linked document on the graph), and vice versa. In link prediction (Bordes et al. 2013; Wang et al. 2021a), the task is to predict the existence or type of an edge between two nodes. This corresponds to our DRP task, where we predict if the given pair of text segments are linked (edge), contiguous (self-loop edge), or random (no edge). Our approach can be viewed as a natural fusion of language-based (e.g. BERT) and graph-based self-supervised learning.

4.3 Strategy to obtain linked documents

As described in §4.1, §4.2, our method *builds* links between documents, and for each anchor segment, *samples* a linked document to put together in the LM input. Here we discuss three key axes to consider to obtain useful linked documents in this process.

Relevance. Semantic relevance is a requisite when building links between documents. If links were randomly built without relevance, LinkBERT would be same as BERT, with simply two options of LM inputs (*contiguous* or *random*). Relevance can be achieved by using hyperlinks or lexical similarity metrics, and both methods yield substantially better performance than using random links (§5.5).

A segment is typically a sentence or a paragraph.

Salience. Besides relevance, another factor to consider (*salience*) is whether the linked document can offer new, useful knowledge that may not be obvious to the current LM. Hyperlinks are potentially more advantageous than lexical similarity links in this regard: LMs are shown to be good at recognizing lexical similarity (Zhang et al. 2020), and hyperlinks can bring in useful background knowledge that may not be obvious via lexical similarity alone (Asai et al. 2020). Indeed, we empirically find that using hyperlinks yields a more performant LM (§5.5).

Diversity. In the document graph, some documents may have a very high in-degree (e.g., many incoming hyperlinks, like the "United States" page of Wikipedia), and others a low in-degree. If we uniformly sample from the linked documents for each anchor segment, we may include documents of high in-degree too often in the overall training data, losing diversity. To adjust so that all documents appear with a similar frequency in training, we sample a linked document with probability inversely proportional to its in-degree, as done in graph data mining literature (Henzinger et al. 2000). We find that this technique yields a better LM performance (§5.5).

5 Experiments

We experiment with our proposed approach in the general domain first, where we pretrain LinkBERT on Wikipedia articles with hyperlinks (§5.1) and evaluate on a suite of downstream tasks (§5.2). We compare with BERT (Devlin et al. 2019) as our baseline. We experiment in the biomedical domain in §6.

5.1 Pretraining setup

Data. We use the same pretraining corpus used by BERT: Wikipedia and BookCorpus (Zhu et al. 2015). For Wikipedia, we use the WikiExtractor to extract hyperlinks between Wiki articles. We then create training instances by sampling *contiguous*, *random*, or *linked* segments as described in §4, with the three options appearing uniformly (33%, 33%, 33%). For BookCorpus, we create training instance by sampling *contiguous* or *random* segments (50%, 50%) as in BERT. We then combine the training instances from Wikipedia and BookCorpus to train LinkBERT. In summary, our pretraining data is the same as BERT, except that we have hyperlinks between Wikipedia articles.

Implementation. We pretrain LinkBERT of three sizes, -tiny, -base and -large, following the configurations of BERT$_{\texttt{tiny}}$ (4.4M parameters), BERT$_{\texttt{base}}$ (110M params), and BERT$_{\texttt{large}}$ (340M params) (Devlin et al. 2019; Turc et al. 2019). We use -tiny mainly for ablation studies.

For -tiny, we pretrain from scratch with random weight initialization. We use the AdamW (Loshchilov & Hutter 2019) optimizer with $(\beta_1, \beta_2) = (0.9, 0.98)$, warm up the learning rate for the first 5,000 steps and then linearly decay it. We train for 10,000 steps with a peak learning rate 5e-3, weight decay 0.01, and batch size of 2,048 sequences with 512 tokens. Training took 1 day on two GeForce RTX 2080 Ti GPUs with fp16.

For -base, we initialize LinkBERT with the BERT$_{\texttt{base}}$ checkpoint released by Devlin et al. (2019) and continue pretraining. We use a peak learning rate 3e-4 and train for 40,000 steps. Other training hyperparameters are the same as -tiny. Training took 4 days on four A100 GPUs with fp16.

For -large, we follow the same procedure as -base, except that we use a peak learning rate of 2e-4. Training took 7 days on eight A100 GPUs with fp16.

Baselines. We compare LinkBERT with BERT. Specifically, for the -tiny scale, we compare with BERT$_{\texttt{tiny}}$, which we pretrain from scratch with the same hyperparameters as LinkBERT$_{\texttt{tiny}}$. The only difference is that LinkBERT uses document links to create LM inputs, while BERT does not.

For -base scale, we compare with BERT$_{\texttt{base}}$, for which we take the BERT$_{\texttt{base}}$ release by Devlin et al. (2019) and continue pretraining it with the vanilla BERT objectives on the same corpus for the same number of steps as LinkBERT$_{\texttt{base}}$.

For -large, we follow the same procedure as -base.

5.2 Evaluation tasks

We fine-tune and evaluate LinkBERT on a suite of downstream tasks.

Extractive question answering (QA). Given a document (or set of documents) and a question as input, the task is to identify an answer span from the document. We evaluate on six popular datasets from the MRQA shared task (Fisch et al. 2019): *HotpotQA* (Yang et al. 2018), *TriviaQA* (Joshi et al. 2017), *NaturalQ* (Kwiatkowski et al. 2019), *SearchQA* (Dunn et al. 2017), *NewsQA* (Trischler et al. 2017), and *SQuAD* (Rajpurkar et al. 2016). As the MRQA shared task does not have a public test set, we split the dev set in half to make new dev and test sets. We follow the fine-tuning method BERT (Devlin et al. 2019) uses for extractive QA. More details are provided in Appendix B.

GLUE. The General Language Understanding Evaluation (GLUE) benchmark (Wang et al. 2018) is a popular suite of sentence-level classification tasks. Following BERT, we evaluate on *CoLA* (Warstadt et al. 2019), *SST-2* (Socher et al. 2013), *MRPC* (Dolan & Brockett 2005), *QQP*, *STS-B* (Cer et al. 2017), *MNLI* (Williams et al. 2017), *QNLI* (Rajpurkar et al. 2016), and *RTE* (Dagan et al. 2005; Haim et al. 2006; Giampiccolo et al. 2007), and report the average score. More fine-tuning details are provided in Appendix B.

5.3 Results

Table 1 shows the performance (F1 score) on MRQA datasets. LinkBERT substantially outperforms BERT on all datasets. On average, the gain is +4.1% absolute for the BERT$_{\texttt{tiny}}$ scale, +2.6% for the BERT$_{\texttt{base}}$ scale, and +2.5% for the BERT$_{\texttt{large}}$ scale. Table 2 shows the results on GLUE, where LinkBERT performs moderately better than BERT. These results suggest that LinkBERT is especially effective at learning knowledge useful for QA tasks (e.g. world knowledge), while keeping performance on sentence-level language understanding.

5.4 Analysis

We further study when LinkBERT is especially useful in downstream tasks.

Improved multi-hop reasoning. In Table 1, we find that LinkBERT obtains notably large gains on QA datasets that require reasoning with multiple documents, such as HotpotQA (+5% over BERT$_{\texttt{tiny}}$), TriviaQA (+6%) and SearchQA (+8%), as opposed to SQuAD (+1.4%) which just has a single document per question. To further gain qualitative insights, we studied in what QA examples LinkBERT succeeds but BERT fails. Figure 3 shows a representative example from HotpotQA. Answering the question needs 2-hop reasoning: identify "Roden Brothers were taken over by Birks Group" from the first document, and then "Birks Group is headquartered in Montreal" from the second document. While BERT tends to simply predict an entity near the question entity ("Toronto" in the first document, which is just 1-hop), LinkBERT correctly predicts the answer in the second document ("Montreal"). Our intuition is that because

https://github.com/attardi/wikiextractor

	HotpotQA	TriviaQA	SearchQA	NaturalQ	NewsQA	SQuAD	Avg.
BERT$_{\texttt{tiny}}$	49.8	43.4	50.2	58.9	41.3	56.6	50.0
LinkBERT$_{\texttt{tiny}}$	**54.6**	**50.0**	**58.6**	**60.3**	**42.8**	**58.0**	**54.1**
BERT$_{\texttt{base}}$	76.0	70.3	74.2	76.5	65.7	88.7	75.2
LinkBERT$_{\texttt{base}}$	**78.2**	**73.9**	**76.8**	**78.3**	**69.3**	**90.1**	**77.8**
BERT$_{\texttt{large}}$	78.1	73.7	78.3	79.0	70.9	91.1	78.5
LinkBERT$_{\texttt{large}}$	**80.8**	**78.2**	**80.5**	**81.0**	**72.6**	**92.7**	**81.0**

Table 1: Performance (F1) on MRQA question answering datasets. LinkBERT consistently outperforms BERT on all datasets across the -tiny, -base, and -large scales. The gain is especially large on datasets that require reasoning with multiple documents in the context, such as HotpotQA, TriviaQA, SearchQA.

	GLUE score
BERT$_{\texttt{tiny}}$	64.3
LinkBERT$_{\texttt{tiny}}$	**64.6**
BERT$_{\texttt{base}}$	79.2
LinkBERT$_{\texttt{base}}$	**79.6**
BERT$_{\texttt{large}}$	80.7
LinkBERT$_{\texttt{large}}$	**81.1**

Table 2: Performance on the GLUE benchmark. LinkBERT attains comparable or moderately improved performance.

	SQuAD	SQuAD distract
BERT$_{\text{base}}$	88.7	85.9
LinkBERT$_{\text{base}}$	**90.1**	**89.6**

Table 3: Performance (F1) on SQuAD when distracting documents are added to the context. While BERT incurs a large drop in F1, LinkBERT does not, suggesting its robustness in understanding document relations.

	HotpotQA	TriviaQA	NaturalQ	SQuAD
BERT$_{\text{base}}$	64.8	59.2	64.8	79.6
LinkBERT$_{\text{base}}$	**70.5**	**66.0**	**70.2**	**82.8**

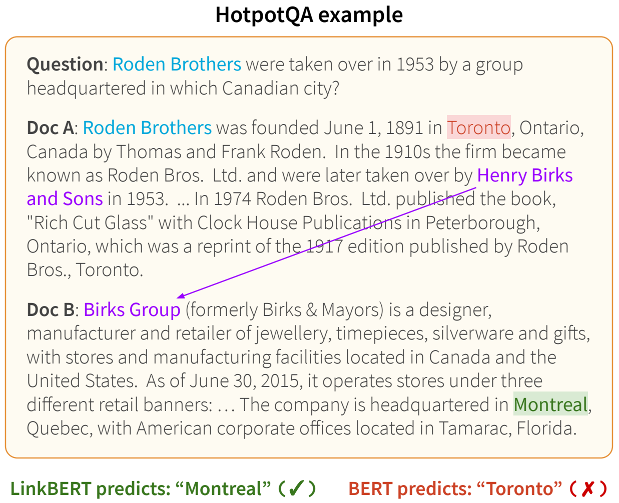
Table 4: Few-shot QA performance (F1) when 10% of fine-tuning data is used. LinkBERT attains large gains, suggesting that it internalizes more knowledge than BERT in pretraining.

	HotpotQA	TriviaQA	NaturalQ	SQuAD
LinkBERT$_{\texttt{tiny}}$	**54.6**	**50.0**	**60.3**	**58.0**
No diversity	53.5	48.0	60.0	57.8
Change hyperlink to TF-IDF	50.0	48.2	59.6	57.6
Change hyperlink to random	49.8	43.4	58.9	56.6

Table 5: Ablation study on what linked documents to feed into LM pretraining (§4.3).

	HotpotQA	TriviaQA	NaturalQ	SQuAD	SQuAD distract
LinkBERT$_{\text{base}}$	**78.2**	**73.9**	**78.3**	**90.1**	**89.6**
No DRP	76.5	72.5	77.0	89.3	87.0

Table 6: Ablation study on the document relation prediction (DRP) objective in LM pretraining (§4.2).


HotpotQA example

Question: Roden Brothers were taken over in 1953 by a group headquartered in which Canadian city?

Doc A: Roden Brothers was founded June 1, 1891 in Toronto, Ontario, Canada by Thomas and Frank Roden. In the 1910s the firm became known as Roden Bros. Ltd. and were later taken over by Henry Birks and Sons in 1953. ... In 1974 Roden Bros. Ltd. published the book, "Rich Cut Glass" with Clock House Publications in Peterborough, Ontario, which was a reprint of the 1917 edition published by Roden Bros., Toronto.

Doc B: Birks Group (formerly Birks & Mayors) is a designer, manufacturer and retailer of jewellery, timepieces, silverware and gifts, with stores and manufacturing facilities located in Canada and the United States. As of June 30, 2015, it operates stores under three different retail banners: ... The company is headquartered in Montreal, Quebec, with American corporate offices located in Tamarac, Florida.

LinkBERT predicts: "Montreal" (✓) BERT predicts: "Toronto" (✗)

Figure 3: Case study of multi-hop reasoning on HotpotQA. Answering the question needs to identify "Roden Brothers were taken over by Birks Group" from the first document, and then "Birks Group is headquartered in Montreal" from the second document. While BERT tends to simply predict an entity near the question entity ("Toronto" in the first document), LinkBERT correctly predicts the answer in the second document ("Montreal").

LinkBERT is pretrained with pairs of linked documents rather than purely single documents, it better learns how to flow information (e.g., do attention) across tokens when multiple related documents are given in the context. In summary, these results suggest that pretraining with linked documents helps for multi-hop reasoning on downstream tasks.

Improved understanding of document relations. While the MRQA datasets typically use ground-truth documents as context for answering questions, in open-domain QA, QA systems need to use documents obtained by a retriever, which may include noisy documents besides gold ones (Chen et al. 2017; Dunn et al. 2017). In such cases, QA systems need to understand the document relations to perform well (Yang et al. 2018). To simulate this setting, we modify the SQuAD dataset by prepending or appending 1–2 distracting documents to the original document given to each question. Table 3 shows the result. While BERT incurs a large performance drop (-2.8%), LinkBERT is robust to distracting documents (-0.5%). This result suggests that pretraining with document links improves the ability to understand document relations and relevance. In particular, our intuition is that the DRP objective helps the LM to better recognize document relations like (anchor document, linked document) in pretraining, which helps to recognize relations like (question, right document) in downstream QA tasks. We indeed find that ablating the DRP objective from LinkBERT hurts performance (§5.5). The strength of understanding document relations also suggests the promise of applying LinkBERT to various retrieval-augmented methods and tasks (e.g. Lewis et al. 2020b), either as the main LM or the dense retriever component.

Improved few-shot QA performance. We also find that LinkBERT is notably good at few-shot learning. Concretely, for each MRQA dataset, we fine-tune with only 10% of the available training data, and report the performance in Table 4. In this few-shot regime, LinkBERT attains more significant gains over BERT, compared to the full-resource regime in Table 1 (on NaturalQ, 5.4% vs 1.8% absolute in F1, or 15% vs 7% in relative error reduction). This result suggests that LinkBERT internalizes more knowledge than BERT during pretraining, which supports our core idea that document links can bring in new, useful knowledge for LMs.

5.5 Ablation studies

We conduct ablation studies on the key design choices of LinkBERT.

What linked documents to feed into LMs? We study the strategies discussed in §4.3 for obtaining linked documents: relevance, salience, and diversity. Table 5 shows the ablation result on MRQA datasets. First, if we ignore relevance and use random document links instead of hyperlinks, we get the same performance as BERT (-4.1% on average; "random" in Table 5). Second, using lexical similarity links instead of hyperlinks leads to 1.8% performance drop ("TF-IDF"). Our intuition is that hyperlinks can provide more salient knowledge that may not be obvious from lexical similarity alone. Nevertheless, using lexical similarity links is substantially better than BERT (+2.3%), confirming the efficacy of placing relevant documents together in the input for LM pretraining. Finally, removing the diversity adjustment in document sampling leads to 1% performance drop ("No diversity"). In summary, our insight is that to create informative inputs for LM pretraining, the linked documents must be semantically relevant and ideally be salient and diverse.

Effect of the DRP objective. Table 6 shows the ablation result on the DRP objective (§4.2). Removing DRP in pretraining hurts downstream QA performance. The drop is large on tasks with multiple documents (HotpotQA, TriviaQA, and SQuAD with distracting documents). This suggests that DRP facilitates LMs to learn document relations.

6 Biomedical LinkBERT (*BioLinkBERT*)

Pretraining LMs on biomedical text is shown to boost performance on biomedical NLP tasks (Beltagy et al. 2019; Lee et al. 2020; Lewis et al. 2020a; Gu et al. 2020). Biomedical LMs are typically trained on PubMed, which contains abstracts and citations of biomedical papers. While prior works only use their raw text for pretraining, academic papers have rich dependencies with each other via citations (references). We hypothesize that incorporating citation links can help LMs learn dependencies between papers and knowledge that spans across them.

With this motivation, we pretrain LinkBERT on PubMed with citation links (§6.1), which we term *BioLinkBERT*, and evaluate on biomedical downstream tasks (§6.2). As our baseline, we follow and compare with the state-of-the-art biomedical LM, PubmedBERT (Gu et al. 2020), which has the same architecture as BERT and is trained on PubMed.

6.1 Pretraining setup

Data. We use the same pretraining corpus used by PubmedBERT: PubMed abstracts (21GB). We use the Pubmed Parser to extract citation links between articles. We then create training instances by sampling *contiguous*, *random*, or *linked* segments as described in §4, with the three options appearing uniformly (33%, 33%, 33%). In summary, our pretraining data is the same as PubmedBERT, except that we have citation links between PubMed articles.

Implementation. We pretrain BioLinkBERT of -base size (110M params) from scratch, following the same hyperparamters as the PubmedBERT$_{\text{base}}$ (Gu et al. 2020). Specifically, we use a peak learning rate 6e-4, batch size 8,192, and train for 62,500 steps. We warm up the learning rate in the first 10% of steps and then linearly decay it. Training took 7 days on eight A100 GPUs with fp16.

Additionally, while the original PubmedBERT release did not include the -large size, we pretrain BioLinkBERT of the -large size (340M params) from scratch, following the same procedure as -base, except that we use a peak learning rate of 4e-4 and warm up steps of 20%. Training took 21 days on eight A100 GPUs with fp16.

Baselines. We compare BioLinkBERT with PubmedBERT released by Gu et al. (2020).

	PubMed- BERT$_{\text{base}}$	BioLink- BERT$_{\text{base}}$	BioLink- BERT$_{\text{large}}$
Named entity recognition			
BC5-chem (Li et al. 2016)	93.33	**93.75**	**94.04**
BC5-disease (Li et al. 2016)	85.62	**86.10**	**86.39**
NCBI-disease (Doğan et al. 2014)	87.82	**88.18**	**88.76**
BC2GM (Smith et al. 2008)	84.52	**84.90**	**85.18**
JNLPBA (Kim et al. 2004)	**80.06**	79.03	**80.06**
PICO extraction			
EBM PICO (Nye et al. 2018)	73.38	**73.97**	**74.19**
Relation extraction			
ChemProt (Krallinger et al. 2017)	77.24	**77.57**	**79.98**
DDI (Herrero-Zazo et al. 2013)	82.36	**82.72**	**83.35**
GAD (Bravo et al. 2015)	82.34	**84.39**	**84.90**
Sentence similarity			
BIOSSES (Soğancıoğlu et al. 2017)	92.30	**93.25**	**93.63**
Document classification			
HoC (Baker et al. 2016)	82.32	**84.35**	**84.87**
Question answering			
PubMedQA (Jin et al. 2019)	55.84	**70.20**	**72.18**
BioASQ (Nentidis et al. 2019)	87.56	**91.43**	**94.82**
BLURB score	81.10	**83.39**	**84.30**

Table 7: Performance on BLURB benchmark. BioLinkBERT attains improvement on all tasks, establishing new state of the art on BLURB. Gains are notably large on document-level tasks such as PubMedQA and BioASQ.

Methods	**Acc.** (%)
BioBERT$_{\text{large}}$ (Lee et al. 2020)	36.7
QAGNN (Yasunaga et al. 2021)	38.0
GreaseLM (Zhang et al. 2022)	38.5
PubmedBERT$_{\text{base}}$ (Gu et al. 2020)	38.1
BioLinkBERT$_{\text{base}}$ (**Ours**)	**40.0**
BioLinkBERT$_{\text{large}}$ (**Ours**)	**44.6**

Table 8: Performance on MedQA-USMLE. BioLinkBERT outperforms all previous biomedical LMs.

6.2 Evaluation tasks

For downstream tasks, we evaluate on the BLURB benchmark (Gu et al. 2020), a diverse set of biomedical NLP datasets, and MedQA-USMLE (Jin et al. 2021), a challenging biomedical QA dataset.

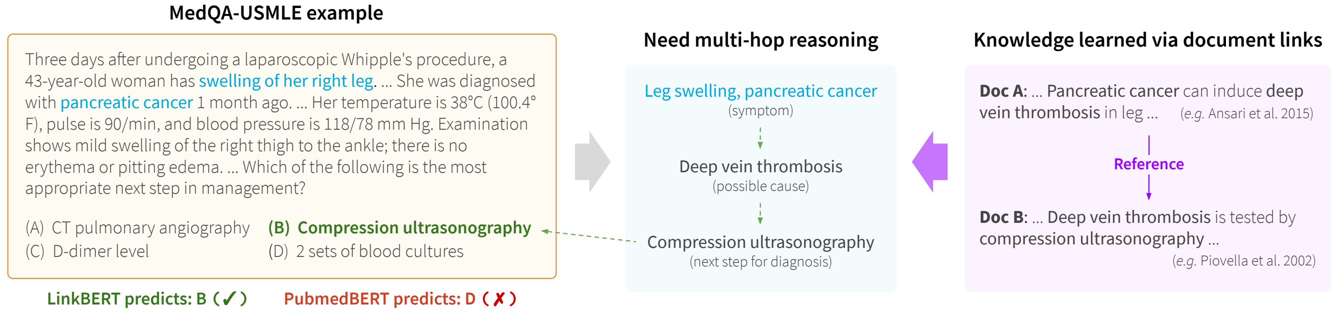
BLURB consists of five named entity recognition tasks, a PICO (population, intervention, comparison, and outcome) extraction task, three relation extraction tasks, a sentence similarity task, a document classification task, and two question answering tasks, as summarized in Table 7. We follow the same fine-tuning method and evaluation metric used by PubmedBERT (Gu et al. 2020).

MedQA-USMLE is a 4-way multi-choice QA task that tests biomedical and clinical knowledge. The questions are from practice tests for the US Medical License Exams (USMLE). The questions typically require multi-hop reasoning, e.g., given patient symptoms, infer the likely cause, and then answer the appropriate diagnosis procedure (Figure 4). We follow the fine-tuning method in Jin et al. (2021). More details are provided in Appendix B.

MMLU-professional medicine is a multi-choice QA task that tests biomedical knowledge and reasoning, and is part of the popular MMLU benchmark (Hendrycks et al. 2021) that is used to evaluate massive language models. We take the BioLinkBERT fine-tuned on the above MedQA-USMLE task, and evaluate on this task without further adaptation.

https://pubmed.ncbi.nlm.nih.gov. We use papers published before Feb. 2020 as in PubmedBERT.

https://github.com/titipata/pubmed_parser



Three days after undergoing a laparoscopic Whipple's procedure, a 43-year-old woman has swelling of her right leg. ... She was diagnosed with pancreatic cancer 1 month ago. ... Her temperature is 38°C (100.4° F), pulse is 90/min, and blood pressure is 118/78 mm Hg. Examination shows mild swelling of the right thigh to the ankle; there is no erythema or pitting edema. ... Which of the following is the most appropriate next step in management?

(A) CT pulmonary angiography (B) Compression ultrasonography
(C) D-dimer level (D) 2 sets of blood cultures

LinkBERT predicts: B (✓)
PubmedBERT predicts: D (✗)
Leg swelling, pancreatic cancer
(symptom)
Deep vein thrombosis
(possible cause)
Compression ultrasonography
(next step for diagnosis)
Reference

Doc A: ... Pancreatic cancer can induce deep vein thrombosis in leg ... (*e.g.* Ansari et al. 2015)

Doc B: ... Deep vein thrombosis is tested by compression ultrasonography ... (*e.g.* Piovella et al. 2002)

Figure 4: Case study of multi-hop reasoning on MedQA-USMLE. Answering the question (left) needs 2-hop reasoning (center): from the patient symptoms described in the question (*leg swelling*, *pancreatic cancer*), infer the cause (*deep vein thrombosis*), and then infer the appropriate diagnosis procedure (*compression ultrasonography*). While the existing PubmedBERT tends to simply predict a choice that contains a word appearing in the question ("blood" for choice D), BioLinkBERT correctly predicts the answer (B). Our intuition is that citation links bring relevant documents together in the same context in pretraining (right), which readily provides the multi-hop knowledge needed for the reasoning (center).

Methods	**Acc.** (%)
GPT-3 (175B params) (Brown et al. 2020)	38.7
UnifiedQA (11B params) (Khashabi et al. 2020)	43.2
BioLinkBERT$_{\text{large}}$ **(Ours)**	**50.7**

Table 9: Performance on MMLU-professional medicine. BioLinkBERT significantly outperforms the largest general-domain LM or QA model, despite having just 340M parameters.

6.3 Results

BLURB. Table 7 shows the results on BLURB. BioLinkBERT$_{\text{base}}$ outperforms PubmedBERT$_{\text{base}}$ on all task categories, attaining a performance boost of +2% absolute on average. Moreover, BioLinkBERT$_{\text{large}}$ provides a further boost of +1%. In total, BioLinkBERT outperforms the previous best by +3% absolute, establishing a new state of the art on the BLURB leaderboard. We see a trend that gains are notably large on document-level tasks such as question answering (+7% on BioASQ and PubMedQA). This result is consistent with the general domain (§5.3) and confirms that LinkBERT helps to learn document dependencies better.

MedQA-USMLE. Table 8 shows the results. BioLinkBERT$_{\text{base}}$ obtains a 2% accuracy boost over PubmedBERT$_{\text{base}}$, and BioLinkBERT$_{\text{large}}$ provides an additional +5% boost. In total, BioLinkBERT outperforms the previous best by +7% absolute, setting a new state of the art. To further gain qualitative insights, we studied in what QA examples BioLinkBERT succeeds but the baseline PubmedBERT fails. Figure 4 shows a representative example. Answering the question (left) needs 2-hop reasoning (center): from the patient symptoms described in the question (*leg swelling*, *pancreatic cancer*), infer the cause (*deep vein thrombosis*), and then infer the appropriate diagnosis procedure (*compression ultrasonography*). We find that while the existing PubmedBERT tends to simply predict a choice that contains a word appearing in the question ("blood" for choice D), BioLinkBERT correctly predicts the answer (B). Our intuition is that citation links bring relevant documents and concepts together in the same context in pretraining (right), which readily provides the multi-hop knowledge needed for the reasoning (center). Combined with the analysis on HotpotQA (§5.4), our results suggest that pretraining with document links consistently helps for multi-hop reasoning across domains (e.g., general documents with hyperlinks and biomedical articles with citation links).

MMLU-professional medicine. Table 9 shows the performance. Despite having just 340M parameters, BioLinkBERT$_{\text{large}}$ achieves 50% accuracy on this QA task, significantly outperforming the largest general-domain LM or QA models such as GPT-3 175B params (39% accuracy) and UnifiedQA 11B params (43% accuracy). This result shows that with an effective pretraining approach, a small domain-specialized LM can outperform orders of magnitude larger language models on QA tasks.

7 Conclusion

We presented LinkBERT, a new language model (LM) pretraining method that incorporates document link knowledge such as hyperlinks. In both the general domain (pretrained on Wikipedia with hyperlinks) and biomedical domain (pretrained on PubMed with citation links), LinkBERT outperforms previous BERT models across a wide range of downstream tasks. The gains are notably large for multi-hop reasoning, multi-document understanding and few-shot question answering, suggesting that LinkBERT effectively internalizes salient knowledge through document links. Our results suggest that LinkBERT can be a strong pretrained LM to be applied to various knowledge-intensive tasks.

Reproducibility

Pretrained models, code and data are available at `https://github.com/michiyasunaga/LinkBERT`.
Experiments are available at
`https://worksheets.codalab.org/worksheets/0x7a6ab9c8d06a41d191335b270da2902e`.

Acknowledgment

We thank Siddharth Karamcheti, members of the Stanford P-Lambda, SNAP and NLP groups, as well as our anonymous reviewers for valuable feedback. We gratefully acknowledge the support of a PECASE Award; DARPA under Nos. HR00112190039 (TAMI), N660011924033 (MCS); Funai Foundation Fellowship; Microsoft Research PhD Fellowship; ARO under Nos. W911NF-16-1-0342 (MURI), W911NF-16-1-0171 (DURIP); NSF under Nos. OAC-1835598 (CINES), OAC-1934578 (HDR), CCF-1918940 (Expeditions), IIS-2030477 (RAPID), NIH under No. R56LM013365; Stanford Data Science Initiative, Wu Tsai Neurosciences Institute, Chan Zuckerberg Biohub, Amazon, JPMorgan Chase, Docomo, Hitachi, Intel, KDDI, Toshiba, NEC, Juniper, and UnitedHealth Group.

For instance, as in Figure 4 (right), Ansari et al. (2015) in PubMed mention that *pancreatic cancer can induce deep vein thrombosis in leg*, and it cites a paper in PubMed, Piovella et al. (2002), which mention that *deep vein thrombosis is tested by compression ultrasonography*. Placing these two documents in the same context yields the complete multi-hop knowledge needed to answer the question ("*pancreatic cancer*" → "*deep vein thrombosis*" → "*compression ultrasonography*").

References

Aghajanyan, A., Okhonko, D., Lewis, M., Joshi, M., Xu, H., Ghosh, G., and Zettlemoyer, L. Htlm: Hyper-text pre-training and prompting of language models. *arXiv preprint arXiv:2107.06955*, 2021.

Ansari, D., Ansari, D., Andersson, R., and Andrén-Sandberg, Å. Pancreatic cancer and thromboembolic disease, 150 years after trousseau. *Hepatobiliary surgery and nutrition*, 4(5):325, 2015.

Asai, A., Hashimoto, K., Hajishirzi, H., Socher, R., and Xiong, C. Learning to retrieve reasoning paths over wikipedia graph for question answering. In *International Conference on Learning Representations (ICLR)*, 2020.

Baker, S., Silins, I., Guo, Y., Ali, I., Högberg, J., Stenius, U., and Korhonen, A. Automatic semantic classification of scientific literature according to the hallmarks of cancer. *Bioinformatics*, 2016.

Beltagy, I., Lo, K., and Cohan, A. Scibert: Pretrained language model for scientific text. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

Bhagavatula, C., Feldman, S., Power, R., and Ammar, W. Content-based citation recommendation. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2018.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L., Goel, K., Goodman, N., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P. W., Krass, M., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X. L., Li, X., Ma, T., Malik, A., Manning, C. D., Mirchandani, S., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J. C., Nilforoshan, H., Nyarko, J., Ogut, G., Orr, L., Papadimitriou, I., Park, J. S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y., Ruiz, C., Ryan, J., Ré, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K., Tamkin, A., Taori, R., Thomas, A. W., Tramèr, F., Wang, R. E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S. M., Yasunaga, M., You, J., Zaharia, M., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., and Liang, P. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., and Yakhnenko, O. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2013.

Bosselut, A., Rashkin, H., Sap, M., Malaviya, C., Çelikyilmaz, A., and Choi, Y. Comet: Commonsense transformers for automatic knowledge graph construction. In *Association for Computational Linguistics (ACL)*, 2019.

Bravo, À., Piñero, J., Queralt-Rosinach, N., Rautschka, M., and Furlong, L. I. Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research. *BMC bioinformatics*, 2015.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Caciularu, A., Cohan, A., Beltagy, I., Peters, M. E., Cattan, A., and Dagan, I. Cross-document language modeling. In *Findings of EMNLP*, 2021.

Calixto, I., Raganato, A., and Pasini, T. Wikipedia entities as rendezvous across languages: Grounding multilingual language models by predicting wikipedia hyperlinks. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2021.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. In *International Workshop on Semantic Evaluation (SemEval)*, 2017.

Chang, W.-C., Yu, F. X., Chang, Y.-W., Yang, Y., and Kumar, S. Pre-training tasks for embedding-based large-scale retrieval. In *International Conference on Learning Representations (ICLR)*, 2020.

Chen, D., Fisch, A., Weston, J., and Bordes, A. Reading wikipedia to answer open-domain questions. In *Association for Computational Linguistics (ACL)*, 2017.

Cohan, A., Feldman, S., Beltagy, I., Downey, D., and Weld, D. S. Specter: Document-level representation learning using citation-informed transformers. In *Association for Computational Linguistics (ACL)*, 2020.

Dagan, I., Glickman, O., and Magnini, B. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*, 2005.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019.

Doğan, R. I., Leaman, R., and Lu, Z. Ncbi disease corpus: a resource for disease name recognition and concept normalization. *Journal of biomedical informatics*, 2014.

Dolan, W. B. and Brockett, C. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*, 2005.

Dunn, M., Sagun, L., Higgins, M., Guney, V. U., Cirik, V., and Cho, K. Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*, 2017.

Fisch, A., Talmor, A., Jia, R., Seo, M., Choi, E., and Chen, D. Mrqa 2019 shared task: Evaluating generalization in reading comprehension. In *Workshop on Machine Reading for Question Answering*, 2019.

Gao, T., Fisch, A., and Chen, D. Making pre-trained language models better few-shot learners. In *Association for Computational Linguistics (ACL)*, 2021.

Giampiccolo, D., Magnini, B., Dagan, I., and Dolan, W. B. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, 2007.

Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., and Poon, H. Domain-specific language model pretraining for biomedical natural language processing. *arXiv preprint arXiv:2007.15779*, 2020.

Guu, K., Lee, K., Tung, Z., Pasupat, P., and Chang, M.-W. Realm: Retrieval-augmented language model pre-training. In *International Conference on Machine Learning (ICML)*, 2020.

Haim, R. B., Dagan, I., Dolan, B., Ferro, L., Giampiccolo, D., Magnini, B., and Szpektor, I. The second pascal recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, 2006.

He, B., Zhou, D., Xiao, J., Jiang, X., Liu, Q., Yuan, N. J., and Xu, T. Integrating graph contextualized knowledge into pre-trained language models. In *Findings of EMNLP*, 2020.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*, 2021.

Henzinger, M. R., Heydon, A., Mitzenmacher, M., and Najork, M. On near-uniform url sampling. *Computer Networks*, 2000.

Herrero-Zazo, M., Segura-Bedmar, I., Martínez, P., and Declerck, T. The ddi corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of biomedical informatics*, 2013.

Hu, W., Liu, B., Gomes, J., Zitnik, M., Liang, P., Pande, V., and Leskovec, J. Strategies for pre-training graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2020.

Jin, D., Pan, E., Oufattole, N., Weng, W.-H., Fang, H., and Szolovits, P. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 2021.

Jin, Q., Dhingra, B., Liu, Z., Cohen, W. W., and Lu, X. Pubmedqa: A dataset for biomedical research question answering. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

Joshi, M., Choi, E., Weld, D. S., and Zettlemoyer, L. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Association for Computational Linguistics (ACL)*, 2017.

Joshi, M., Chen, D., Liu, Y., Weld, D. S., Zettlemoyer, L., and Levy, O. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 2020.

Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-t. Dense passage retrieval for open-domain question answering. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

Khadka, A., Cantador, I., and Fernandez, M. Exploiting citation knowledge in personalised recommendation of recent scientific publications. In *Language Resources and Evaluation Conference (LREC)*, 2020.

Khashabi, D., Khot, T., Sabharwal, A., Tafjord, O., Clark, P., and Hajishirzi, H. Unifiedqa: Crossing format boundaries with a single qa system. In *Findings of EMNLP*, 2020.

Kim, J.-D., Ohta, T., Tsuruoka, Y., Tateisi, Y., and Collier, N. 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*, 2004.

Krallinger, M., Rabal, O., Akhondi, S. A., Pérez, M. P., Santamaría, J., Rodríguez, G. P., Tsatsaronis, G., and Intxaurrondo, A. Overview of the biocreative vi chemical-protein interaction track. In *Proceedings of the sixth BioCreative challenge evaluation workshop*, 2017.

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics (TACL)*, 2019.

Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C. H., and Kang, J. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 2020.

Levine, Y., Wies, N., Jannai, D., Navon, D., Hoshen, Y., and Shashua, A. The inductive bias of in-context learning: Rethinking pretraining example design. *arXiv preprint arXiv:2110.04541*, 2021.

Lewis, P., Ott, M., Du, J., and Stoyanov, V. 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*, 2020a.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020b.

Li, J., Sun, Y., Johnson, R. J., Sciaky, D., Wei, C.-H., Leaman, R., Davis, A. P., Mattingly, C. J., Wiegers, T. C., and Lu, Z. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. *Database*, 2016.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Ma, Z., Dou, Z., Xu, W., Zhang, X., Jiang, H., Cao, Z., and Wen, J.-R. Pre-training for ad-hoc retrieval: Hyperlink is also you need. In *Conference on Information and Knowledge Management (CIKM)*, 2021.

Margolis, E., Laurence, S., et al. *Concepts: core readings*. Mit Press, 1999.

Nentidis, A., Bougiatiotis, K., Krithara, A., and Paliouras, G. Results of the seventh edition of the bioasq challenge. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 2019.

Nye, B., Li, J. J., Patel, R., Yang, Y., Marshall, I. J., Nenkova, A., and Wallace, B. C. A corpus with multi-level annotations of patients, interventions and outcomes to support language processing for medical literature. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, 2018.

Oguz, B., Chen, X., Karpukhin, V., Peshterliev, S., Okhonko, D., Schlichtkrull, M., Gupta, S., Mehdad, Y., and Yih, S. Unified open-domain question answering with structured and unstructured knowledge. *arXiv preprint arXiv:2012.14610*, 2020.

Petroni, F., Rocktäschel, T., Lewis, P., Bakhtin, A., Wu, Y., Miller, A. H., and Riedel, S. Language models as knowledge bases? In *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

Piovella, F., Crippa, L., Barone, M., D'Angelo, S. V., Serafini, S., Galli, L., Beltrametti, C., and D'Angelo, A. Normalization rates of compression ultrasonography in patients with a first episode of deep vein thrombosis of the lower limbs: association with recurrence and new thrombosis. *Haematologica*, 87(5):515–522, 2002.

Qazvinian, V. and Radev, D. R. Scientific paper summarization using citation summary networks. In *International Conference on Computational Linguistics (COLING)*, 2008.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research (JMLR)*, 2020.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. Squad: 100,000+ questions for machine comprehension of text. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2016.

Seonwoo, Y., Lee, S.-W., Kim, J.-H., Ha, J.-W., and Oh, A. Weakly supervised pre-training for multi-hop retriever. In *Findings of ACL*, 2021.

Sheng, E., Chang, K.-W., Natarajan, P., and Peng, N. Towards controllable biases in language generation. In *the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)-Findings, long*, 2020.

Smith, L., Tanabe, L. K., nee Ando, R. J., Kuo, C.-J., Chung, I.-F., Hsu, C.-N., Lin, Y.-S., Klinger, R., Friedrich, C. M., Ganchev, K., et al. Overview of biocreative ii gene mention recognition. *Genome biology*, 2008.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2013.

Soğancıoğlu, G., Öztürk, H., and Özgür, A. Biosses: a semantic sentence similarity estimation system for the biomedical domain. *Bioinformatics*, 2017.

Sun, T., Shao, Y., Qiu, X., Guo, Q., Hu, Y., Huang, X.-J., and Zhang, Z. Colake: Contextualized language and knowledge embedding. In *International Conference on Computational Linguistics (COLING)*, 2020.

Trischler, A., Wang, T., Yuan, X., Harris, J., Sordoni, A., Bachman, P., and Suleman, K. Newsqa: A machine comprehension dataset. In *Workshop on Representation Learning for NLP*, 2017.

Turc, I., Chang, M.-W., Lee, K., and Toutanova, K. Well-read students learn better: On the importance of pre-training compact models. *arXiv preprint arXiv:1908.08962*, 2019.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, 2018.

Wang, H., Ren, H., and Leskovec, J. Relational message passing for knowledge graph completion. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021a.

Wang, X., Gao, T., Zhu, Z., Zhang, Z., Liu, Z., Li, J., and Tang, J. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 2021b.

Warstadt, A., Singh, A., and Bowman, S. R. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 2019.

Williams, A., Nangia, N., and Bowman, S. R. A broad-coverage challenge corpus for sentence understanding through inference. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2017.

Xie, T., Wu, C. H., Shi, P., Zhong, R., Scholak, T., Yasunaga, M., Wu, C.-S., Zhong, M., Yin, P., Wang, S. I., Zhong, V., Wang, B., Li, C., Boyle, C., Ni, A., Yao, Z., Radev, D., Xiong, C., Kong, L., Zhang, R., Smith, N. A., Zettlemoyer, L., and Yu, T. Unifiedskg: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. *arXiv preprint arXiv:2201.05966*, 2022.

Yang, B., Yih, W.-t., He, X., Gao, J., and Deng, L. Embedding entities and relations for learning and inference in knowledge bases. In *International Conference on Learning Representations (ICLR)*, 2015.

Yang, Z., Qi, P., Zhang, S., Bengio, Y., Cohen, W. W., Salakhutdinov, R., and Manning, C. D. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

Yasunaga, M., Zhang, R., Meelu, K., Pareek, A., Srinivasan, K., and Radev, D. Graph-based neural multi-document summarization. In *Conference on Computational Natural Language Learning (CoNLL)*, 2017.

Yasunaga, M., Kasai, J., Zhang, R., Fabbri, A. R., Li, I., Friedman, D., and Radev, D. R. Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

Yasunaga, M., Ren, H., Bosselut, A., Liang, P., and Leskovec, J. QA-GNN: Reasoning with language models and knowledge graphs for question answering. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2021.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations (ICLR)*, 2020.

Zhang, X., Bosselut, A., Yasunaga, M., Ren, H., Liang, P., Manning, C. D., and Leskovec, J. Greaselm: Graph reasoning enhanced language models for question answering. In *International Conference on Learning Representations (ICLR)*, 2022.

Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., and Liu, Q. Ernie: Enhanced language representation with informative entities. *arXiv preprint arXiv:1905.07129*, 2019.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *International Conference on Computer Vision (ICCV)*, 2015.

A Ethics, limitations and risks

We outline potential ethical issues with our work below. First, DRAGON is trained on the same text corpora (e.g., Wikipedia, Books, PubMed) as in existing language models. Consequently, DRAGON could reflect the same biases and toxic behaviors exhibited by language models, such as biases about race, gender, and other demographic attributes (Sheng et al. 2020).

Another source of ethical concern is the use of the MedQA-USMLE evaluation (Jin et al. 2021). While we find this clinical reasoning task to be an interesting testbed for DRAGON and for multi-hop reasoning in general, we do not encourage users to use the current models for real world clinical prediction.

B Fine-tuning details

We apply the following fine-tuning hyperparameters to all models, including the baselines.

MRQA. For all the extractive question answering datasets, we use `max_seq_length` = 384 and a sliding window of size 128 if the lengths are longer than `max_seq_length`.

For the -tiny scale ($\text{BERT}_{\texttt{tiny}}$, $\text{LinkBERT}_{\texttt{tiny}}$), we choose learning rates from {5e-5, 1e-4, 3e-4}, batch sizes from {16, 32, 64}, and fine-tuning epochs from {5, 10}.

For -base ($\text{BERT}_{\texttt{base}}$, $\text{LinkBERT}_{\texttt{base}}$), we choose learning rates from {2e-5, 3e-5}, batch sizes from {12, 24}, and fine-tuning epochs from {2, 4}.

For -large ($\text{BERT}_{\texttt{large}}$, $\text{LinkBERT}_{\texttt{large}}$), we choose learning rates from {1e-5, 2e-5}, batch sizes from {16, 32}, and fine-tuning epochs from {2, 4}.

GLUE. We use `max_seq_length` = 128.

For the -tiny scale ($\text{BERT}_{\texttt{tiny}}$, $\text{LinkBERT}_{\texttt{tiny}}$), we choose learning rates from {5e-5, 1e-4, 3e-4}, batch sizes from {16, 32, 64}, and fine-tuning epochs from {5, 10}.

For -base and -large ($\text{BERT}_{\texttt{base}}$, $\text{LinkBERT}_{\texttt{base}}$, $\text{BERT}_{\texttt{large}}$, $\text{LinkBERT}_{\texttt{large}}$), we choose learning rates from {5e-6, 1e-5, 2e-5, 3e-5, 5e-5}, batch sizes from {16, 32, 64} and fine-tuning epochs from 3–10.

BLURB. We use `max_seq_length` = 512 and choose learning rates from {1e-5, 2e-5, 3e-5, 5e-5, 6e-5}, batch sizes from {16, 32, 64} and fine-tuning epochs from 1–120.

MedQA-USMLE. We use `max_seq_length` = 512 and choose learning rates from {1e-5, 2e-5, 3e-5}, batch sizes from {16, 32, 64} and fine-tuning epochs from 1–6.