LEVERAGING EXTERNAL KNOWLEDGE RESOURCES TO ENABLE DOMAIN-SPECIFIC COMPREHENSION

Anonymous authors
Paper under double-blind review

ABSTRACT

Machine Reading Comprehension (MRC) has been a long-standing problem in NLP and, with the recent introduction of the BERT family of Transformer based Language Models (TLM), it has come a long way to getting solved. Unfortunately, however, when BERT variants trained on general text corpora are applied to domain-specific (DS) text, their performance inevitably degrades on account of the *domain shift* i.e. genre/subject matter discrepancy between the training and downstream application data. Knowledge graphs (KGs) act as reservoirs for either open or closed domain information and prior studies have shown that they can be used to improve the performance of general-purpose TLMs in DS applications. Building on existing work, we introduce a method using Multi-Layer Perceptrons (MLPs) for aligning and integrating embeddings extracted from KGs (KGE) with TLM embedding spaces. We fuse the aligned KGE with general-purpose TLMs and fine-tune them for two MRC tasks viz. span detection (COVID-QA) and multiple-choice questions (PubMedQA). On the COVID-QA dataset, we see that our approach allows general-purpose TLMs to perform similarly to their DS counterparts, mostly in terms of EM (Exact Match) but still recognizable with regards to F1. With regards to PubMedQA, we observe an overall improvement in accuracy for general purpose TLMs while the F1 stays relatively the same over the DS models.

1 INTRODUCTION

Machine Reading Comprehension (MRC) is defined as a class of supervised question answering (QA) problems wherein a system learns a function to answer a question given an associated passage(s), i.e. the answer to the question *exists* [1] in the passage itself and the system must retrieve it. Mathematically, $MRC\colon f(C, Q) \rightarrow A$ where C is the relevant context, Q is the question and A is the answer space to be learned (Liu et al., 2019).

MRC is one of the most challenging areas of NLP since a system needs to deal with multiple facets - viz. identifying entities, supporting facts in context, the intent of the question, etc. - to answer correctly. Fortunately, with the introduction of the Transformer (Vaswani et al., 2017) model and subsequent BERT (Devlin et al., 2019) family of models (Rogers et al., 2020), the state-of-the-art (SOTA) in MRC has moved forward by leaps and bounds.

Generally, TLMs are trained on enormous volumes of open-domain text such as Wikipedia or book corpora, which allows them to produce extremely rich word representations. This phase is known as *pre-training* (PT) and takes a considerable amount of time and computational resources. A pre-trained TLM can be further trained to perform a more specific task (*inductive transfer learning*) on a new dataset by adding a task-specific layer on top of the PT architecture. This phase is called *fine-tuning* (FT) and is more computationally lightweight, enabling a wider range of practitioners to enter the field.

The standard *recipe* of PT + FT (Linzen, 2020) has thus far served the NLP community well with scores on the GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019) benchmarks standing as testament. However, the performance of TLMs tends to degrade when the downstream task data belongs to a different or more specific domain than its PT counterpart, i.e. TLMs find it difficult to generalize *into-the-wild*. That being said, we recognize the idiosyncrasies of every domain and appreciate how difficult it must be for a general-purpose model like BERT to answer esoteric questions such as, "*What is IFITM?*"[2].

A quintessential resource for accessing DS information (or Domain Knowledge (DK)) are knowledge graphs (KG) – large, structured graphs of facts stored as triples (entity-relation-entity). Previous studies (cf. sec. 2) have shown how LMs can leverage KGs to improve performance on various NLP tasks. Besides query expansion, KGs also allow

[1] We exclude newer trends such as unanswerable questions, conversational questions, etc.

[2] Interferon-Induced Transmembrane

for the training of embeddings (KGE), which aim to capture the semantics of an entity and its behaviour with other entities, as described by the relations in the KG.

Given that KGE provides valuable information about entities, we wanted to incorporate them into our question representations. However, as pointed out by Liu et al. (2020), external embeddings cannot be directly fed to the input signal of a TLM because of the different criteria (objective function) used in generating them (KGE & TLM embeddings) and concatenating them would lead to inconsistent or *heterogeneous* embedding spaces (HES). As such, we propose a simple *embedding homogenization* technique, inspired by the recent interest in feedforward neural networks (FFNN) (Liu et al., 2021), to fuse entity KGE into the question representation during the FT phase for MRC. Our proposed technique does not rely on a vocabulary overlap between the domains to be aligned, unlike the Mikolov et al. (2013) approach. It can leverage the full array of information provided by the KG, not just information for entities present in both vocabularies.

Additionally, we also make use of *definition embeddings* (DE) for the identified domain terms (terms in the text that have a KG counterpart) to provide additional knowledge. Consider the domain term, *Reduced* with the definition, *Made less in size or amount or degree*. Such definitions are passed through our TLMs, in feature extraction mode, and the *pooler* output (cf. sec. 3.5) is treated as the DE. We hypothesized that KGE alone would not be enough to see appreciable gains and thus, additional DS vectors representing what a concept means *should* provide the extra boost.

Finally, we apply our proposed method to two MRC tasks, span extraction and multiple-choice questions. The former was in the form of the COVID-QA dataset (Möller et al., 2020) which we spent considerable time cleaning and set new benchmarks on using a like-for-like RoBERTa model. The latter involved applying our method to PubMedQA (Jin et al., 2019b), a collection of biomedical oriented yes/no/maybe questions.

A fair question, given that biomedical variants of BERT exist, such as SciBERT (Beltagy et al., 2019) and BioBERT (Lee et al., 2020), is, why would we want to adapt vanilla BERT to the given domain at hand? First, BERT trained on general-purpose corpora seems to learn artefacts of the language it was trained on that may be useful for question answering. Second, full PT on a DS corpus may be infeasible or ineffective if that corpus is small. Third, if the DS corpus is large, then the PT gets expensive. Thus, we sought to synthesize an approach that could equip regular BERT with the information necessary to perform closer to the level of DS models. We seek to show that PT over large amounts of DS text is *not the only way* to realize powerful DS models.

We summarize our contributions as follows:

- We propose a domain-agnostic strategy for alignment across embeddings spaces using FFNNs that i) **does not hinge on vocabulary overlap** and ii) **can be used even if the domain terms are phrases or pseudo-words** (ex. `entry-cluster` or `related_to`).
- We show that BERT can perform very similar to DS models (Bio/Sci-BERT), using our modifications, despite the different data used during PT.
- We also release [3] an updated (i.e. cleaned) version of the COVID-QA dataset.

2 Related Work

Recent studies have attempted to incorporate additional information into TLMs either during PT/FT by making modifications to either the input or output representation or through additional layers to the base TLM, which takes care of the knowledge integration. When modifying the input, K-BERT (Liu et al., 2020) uses external information in the form of *raw text*. They expand the identified entities in the input sentence with one-hop KG-triples and FT BERT using the updated representation and *soft-position embeddings* to account for the distortion in the updated text. E-BERT Poerner et al. (2020) provides additional signals, alongside the question representation, in the form of Wikipedia embeddings that have been learned and homogenized to the transformer space. As our work is focused on FT and updating the input representations, we believe that a discussion of the other two sides of knowledge integration is beyond the scope of this study and refer readers to Colon-Hernandez et al. (2021) for a survey.

2.1 Text Integration

In text integration, external information is added to the input side in the form of *raw strings*. Usually, these texts are either term definitions or triples from a KG. Yu et al. (2021) injects dictionary definitions for rare words into the input representation and PT using a custom objective function, consisting of the regular MLM (masked language modelling)

[3] The code & dataset will be released on publication.

loss, maximization of mutual information between terms and their definition and definition discrimination i.e. whether the appended definition refers to a rare or regular word. Ingenious as this approach might be, it suffers from the drawback of additional PT with custom loss functions, which we are trying to avoid at all costs.

2.1.1 KNOWLEDGE TRIPLES

The more popular way of integrating plain text knowledge is via a KG. KG-triples can be defined as information three-tuples (`<subject, predicate, object>`) described in a condensed *pseudo-language*. For example, a KG-triple describing the fact *a rock is heavier than paper* might look like `heavier_than(rock, paper)` or `(rock, heavier_than, paper)`.

One of the earliest attempts at integrating KG-triples was COMET (Bosselut et al., 2019), a GPT (Radford et al., 2018) model trained on the subject and predicate tokens of the triples with the training objective being the prediction of the object token. The model showed to a certain degree that PT allows for the capture of commonsense information. PT in this manner can be considered computationally efficient w.r.t DS TLMs such as Bio/Sci-BERT owing to the number of training samples used ("100K triples for learning 34 relation types for their ConceptNet experiments, and 710K training triples for ATOMIC" (Bouraoui et al., 2020)) while semantically suboptimal since it is trained solely on triples that are expressed in pseudo-language, which limits the ability of the model to generalize to well-formed natural text. COMET is set apart from other TLMs in that it directly uses the KG-triples as corpus instead of augmenting the input sequences.

Lv et al. (2020) present a method for integrating KG-triples in a multiple-choice question answering (MCQA) setting. First, entities in both the question and each answer choice are identified. Next, they query ConceptNet to identify paths between question and answer choice entities that are less than "3 hops" apart. An input record is created for each answer choice, as is common in MCQA, and KG-triples connecting question entities and answer choice entities are converted to *pseudo sentences* before being prepended to the input record. For example, the KG-triple `(rock, heavier_than, paper)` would be converted to the *pseudo sentence* "rock heavier than paper" and then added to the input.

2.2 EMBEDDING INTEGRATION

A more powerful method of knowledge augmentation is by using *external embeddings* trained on a relevant domain. For such integration, information is added to the input sequence after it is converted to an embedding form. Poerner et al. (2020) propose E-BERT, a BERT model FT using Wikipedia2Vec (Yamada et al., 2016) embeddings for LAMA (Petroni et al., 2019), a cloze-style unsupervised QA task. They homogenize their entity embeddings using the Mikolov cross-lingual alignment strategy. Results indicate the superiority of their approach. However, one of the fundamental requirements of their method was in homogenizing using an approach that relies on a vocabulary overlap to learn a strong mapping weight matrix. In domains such as ours where we do not see a huge overlap, such an alignment technique does not fare well.

Sharma et al. (2019) propose an intelligent framework for utilizing KGE for the biomedical inference task, MedNLI (Romanov & Shivade, 2018) i.e. classifying a sentence pair (premise, hypothesis) as entailment, neutral, or contradiction. Although their task is not QA and does not use TLMs, we study the way they utilized DK. After identifying all the biomedical concepts in their dataset, they create a sub-graph from the UMLS KG (Unified Medical Language System) (Bodenreider, 2004) on which they train DistMult (Yang et al., 2014) embeddings. These embeddings are then concatenated with DS BioELMo (Jin et al., 2019a) embeddings and used with the SOTA NLI architecture, at the time, ESIM (Chen et al., 2017). Although they directly concatenated external embeddings, they trained the entire ESIM model to project these vectors in the same space. In doing so, they circumvent the HES issue.

2.3 EMBEDDING ALIGNMENT

Mikolov et al. (2013) provided a strategy for cross-lingual embedding alignment for the task of translation, using a linear objective function presented in eq. 1, where $\{(x_i, z_i)\}$ are a set of n translation pairs - x_i, the *source* and z_i the *target* embedding. Once W is learned, the target embedding z_i for a given source embedding x_i is computed simply as $z_i = Wx_i$. Xing et al. (2015) realized that while the word vectors themselves were trained using a log-likelihood objective (skip-gram), the transformation matrix was learned using a mean squared loss objective. As such, they rectified the equation by enforcing an orthogonality constraint $\mathbf{W^T W} = \mathbf{I}$ and normalizing all vectors to unit length.

Later, Zhang et al. (2019) showed that this constraint fails for languages that are *non-isomorphic* i.e. language pairs where a word in the source language does not map to a single word in the target language and vice versa. To alleviate this issue, they enforced constraints of *length and center invariance* i.e. monolingual embeddings ($\mathbf{x}_i$, n embeddings

in total) must be of unit length ($\|\mathbf{x}_i\|_2 = 1, 1 \leq i \leq n$) and mean of all embeddings must be 0 ($\sum_{i=1}^{n} \mathbf{x}_i = \mathbf{0}$). Their method, known as *iterative normalization*, when applied to three cross-lingual word embedding (CWE) models, was found to improve performance on translation tasks between English and seven other languages.

$$\min_{W} \sum_{i=1}^{n} \|Wx_i - z_i\|^2 \tag{1}$$

While all of these modifications address different shortcomings of the base linear transformation strategy, they *still* depend on two key criteria: 1) a mapping dictionary and 2) a common dimensionality between *source* and *target* embeddings. The first criterion is more difficult to satisfy today than when the Mikolov approach was first introduced. The vocabulary of modern TLMs does not consist of complete words, but rather word *tokens* that often do not constitute a word on their own. This poses a challenge when trying to integrate embeddings from a knowledge base, for example, where embeddings correspond to whole words or even word phrases. This is compounded by the second criterion which requires the embeddings to be of the same dimension. When working with embeddings from two TLMs, this is often the case, but less common when working with embeddings from other resources. Such criteria significantly limit the utility of the previous approaches.

As such, we decided to use the original Mikolov strategy as a baseline for two reasons, i) it avoids the similar dimensionality constraint & ii) it was used by a recently published knowledge enhanced BERT variant (E-BERT).

3 Proposed Methodology

The overall pipeline can be broken down into four phases - 1) **Entity linking** 2) **KGE homogenization** 3) **DE generation** and 4) **FT with external knowledge infusion**. Each phase is described below along with the resources used and the steps taken to clean COVID-QA.

3.1 Resources Used

The four resources used for this project were the MRC datasets (COVID-QA and PubMedQA), **pretrained UMLS KGE** (Maldonado et al., 2019), UMLS Metathesaurus and an entity linker to the UMLS (MetaMap). Given that the domain is COVID-19, KGs pertaining to it, such as those developed by Domingo-Fernández et al. (2021) and Wise et al. (2020) would seem to be more relevant than UMLS. However, our choice for using the UMLS was twofold; first, a well-maintained entity linker is available, which reduces the uncertainties in that process, and, second, UMLS has been maintained for a very long time and is thus robust in terms of coverage and accuracy.

1. **COVID-QA** As mentioned previously, we perform span detection on the recently released COVID-QA, a SQuAD (Rajpurkar et al., 2018) style dataset with 2019 question-answer pairs based on 147 scientific articles drawn from the CORD-19 dataset (Wang et al., 2020) and annotated by 15 experts in the biomedical field. COVID-QA differs from SQuAD on three accounts: 1) specialized domain (COVID-19), 2) longer articles "(6118.5 vs 153.2 tokens)" and 3) longer answer spans "(13.9 vs. 3.2 words)".
2. **PubMedQA** Multiple-choice QA was performed on the PubMedQA benchmark which has a collection of 1k expert-annotated instances of yes/no/maybe answer biomedical questions. There also exists around 270k unannotated samples. As such, we rely on the labelled examples for the task. The statistics of the labelled dataset reveal an unbalanced distribution of answers with the majority leaning towards, yes (55.2%), followed by no (33.8%) and maybe (11.0%). Furthermore, the 1k samples are split into 500 samples each of training and test with the training data being further broken up into 10-folds with 450 samples in training and 50 in the development set.
3. **UMLS** is composed of three knowledge sources:**Metathesaurus** (MT), **Semantic Network** (SN), and **SPECIALIST Lexicon and Lexical Tools**. The MT is a massive collection of information on various biomedical concepts including concept names, definitions, hierarchies, relations to other concepts, etc., derived from vocabularies such as MeSH and SNOMED CT. The latest release [4] contains over 4 million concepts and 10 million concept names i.e. multiple concepts might be mapped to the same *concept type*. For example, concept names *main* and *principal* might get mapped to the concept *primary*.

   We used the UMLS to extract entity definitions from the **MRDEF** table of the MT. The other two sources were not used. Overall, the latest downloadable version of the MT contains 296,789 concept definitions.

[4] https://uts.nlm.nih.gov/uts/umls/home

4. **Pre-Trained UMLS KGE** We used the PT-UMLS-KGE (Maldonado et al., 2019). They claimed that traditional KGE algorithms such as TransE (Bordes et al., 2013) and DistMult (Yang et al., 2014) are ineffective at embedding the UMLS because they "assume a single knowledge graph" whereas UMLS consists of two interrelated graphs, MT and SN. Adapting the KBGAN model (Cai & Wang, 2018) to the UMLS, they trained KGE by feeding UMLS-triples to two separate generators (one each for MT and SN) and a single discriminator. F1 scores from their downstream application of two related binary classification tasks showcased the effectiveness of incorporating their trained KGE as opposed to their absence. In total, they provide around 3.2M 50-dimension entity embeddings.
5. **MetaMap** Following Sharma et al. (2019), we use MetaMap (Aronson & Lang, 2010) as our entity identifier. MetaMap works in tandem with UMLS since it breaks down input sentences according to the UMLS concepts it discovers within it. While it provides a range of information for an extracted concept, such as its preferred name, semantic type, etc., we were only interested in working with the *preferred name* as they were considered as our *knowledge graph entities*.

3.2 Preprocessing (COVID-QA cleanup)

In its current state, COVID-QA was rife with syntactical and encoding issues. We focussed mainly on cleaning the questions rather than the associated passages since i) they were recently curated, and ii) we wanted to add the extra signal there, and as such, we needed as clean as an input as possible. To check for grammatical and semantic errors in the text, the use of a search engine such as Google would have been ideal. Unfortunately, there are not many packages that allow text correction via search engines and those that do either have a cap on the possible daily hits and/or charge a fee for their services. As such, we decided to run the questions through Grammarly [5], which has equally impressive text rectification capabilities and manually fixed the questions. In total, 1020 questions (50.5%) were cleaned. The following issues were identified and fixed accordingly,

- Excess spaces (*For what sca algorithm was applied to improve the anfis model ?*)
- Missing spaces (*What percentage of patients do not return for* ***followup*** *after hiv testing?*)
- Uncapitalized acronyms (*What is* ***ifitm****?*)
- Repeated words (***Was was*** *the sample size?*)
- Spelling mistakes (*How does* ***mannanose*** *binding lectin (mbl) affect elimination of hiv-1 pathogen?*)
- Grammatical issues (*What suggests that ip-10 plays a significant role* ***on*** *the pathogenesis of pneumonia?*)

Finally, we ran NLM's `replace_UTF8` [6] program to, as the name suggests, replace all the unicode characters. Thus, questions such as,

> *Why is the phage displaying an scf.v against* β***-amyloid*** *fibrils is a good diagnostic for alzheimers and parkinson's disease?*

became

> *Why is the phage displaying an scf.v against* ***beta-amyloid*** *fibrils is a good diagnostic for Alzheimer's and Parkinson's disease?*

3.3 Entity Linking

We ran MetaMap on the 2019 COVID-QA and 1k PubMedQA questions revealing 1897 and 2782 entities respectively. Among them, 1837 and 2694 were present in the PT-KGE. Finally, only 1452 and 2078 entities had definitions in the MT and were chosen for homogenization. MetaMap had the following settings: suppress all numeric concepts (–*no_nums ALL*), only unique acronyms (*-u*) and no derivational variants (*-d*). These settings led to the least noise in the linking process and ultimately best performance for our models. We also use a custom acronym and concept exclusion list, which was curated after studying the vanilla outputs from MetaMap. Numerical concepts were excluded from the linking process since it is known that BERT does not deal with numbers very well (Wallace et al., 2019).

[5] https://www.grammarly.com/
[6] https://lhncbc.nlm.nih.gov/ii/tools/MetaMap/additional-tools/ReplaceUTF8.html

3.4 KGE HOMOGENIZATION

According to the **universal approximation theorem** (Hornik et al., 1989), an MLP with sigmoid activation and at least one hidden layer of arbitrary length can approximate any well behaved function. Using this as a guiding principal, we propose a method for learning homogenized $\mathbb{R}^{d^{TLM}}$ vectors from $\mathbb{R}^{d^{KGE}}$ ones, where $d^{TLM} = 768$ and $d^{KGE} = 50$.

The FFNN used to homogenize the KGE consisted of only *one hidden layer*. Inputs to the network were first subjected to a *dropout* regularization, with a probability of 0.25, following which they passed through the hidden layer ($d^{hidden} = 300$). Outputs from the hidden layer were processed via a *layer normalization* and **TanH** activation. A final linear transformation was applied to the outputs to produce a $\mathbb{R}^{d^{TLM}}$ vector. We chose TanH for the hidden layer activation because it restricts values between $[-1, 1]$, a characteristic observed in the elements of TLM embeddings in general. The network was trained with a batch size of 256 for 30 epochs and optimized using Adam with an L2 penalty of 0.001.

Our objective function was to minimize the **MSE** loss between the FFNN output and the *average of the entity subword embeddings*, obtained from the TLM. Consider the entity *cysteine*. When tokenized using BERT, it generates the subwords, [*'cy', '##stein', '##e'*]. An average of these token embeddings, obtained using BERT's vocabulary lookup table, represents the *target* output for the entity.

We selected 10k samples from PT-KGE to train our projection network. Once the network is trained, the final linear transformation stands as the homogenized embeddings for our KGE.

3.5 DEFINITION EMBEDDINGS

We hypothesized that homogenized KGE alone would not be enough to see significant performance gains on the task. Thus, we decided to incorporate *entity definitions* to provide an added source of external knowledge. However, simply using the text form of the definitions would create longer questions and shorter context representations, which in turn would require more negative samples to be generated since the BERT-family of TLMs can only handle 512 input tokens.

We vectorized the definitions by passing them through the respective TLMs, in a *feature extraction mode*, and using the model-specific *pooler output*, which for the BERT-family is the $[CLS]$ token further processed by a linear layer (weights obtained from the NSP task) and TanH activation. We usually observed that it is more beneficial to use the processed $[CLS]$ embedding i.e. the pooler output rather than the regular $[CLS]$ and thus decided to use the former.

3.6 FT WITH IMPROVED QUESTION REPRESENTATION

For each entity, we thus had two embeddings viz. a homogenized KGE and a DE. We observed increased FT time and subpar results when we added the embeddings *separately* into the input representation, by the schemes described below. As such, we decided to *average* the two embeddings, to form the final external knowledge vector. Two schemes of integrating these embeddings were explored and are described below. Both schemes are explained using the sample question, *What is the main cause of HIV-1 infection in children?*, which when tokenized using BERT's tokenizer yields the following set of tokens, *['what', 'is', 'the', 'main', 'cause', 'of', 'hiv', '-', '1', 'infection', 'in', 'children', '?']*. We FT our models in the regular fashion, i.e. we do not make architectural changes to the model.

1. **BERTRAM concatenation**: According to Schick & Schütze (2020), the external embedding for a given token should be concatenated *alongside* it using the **/** (slash) symbol as a separator. Thus, the representation for the above question would become:

   [CLS] [what] [is] [the] [main] [/] [main] [cause] [/] [cause] [of] [hiv 1 infection] [/] [hiv] [1] [infection] [in] [children] [/] [children] [?] [SEP] [context tokens]

   where the terms in blue are the non-entity terms, terms in red are the identified entities and the ones in violet are their corresponding external knowledge embeddings.

2. **DEKCOR concatenation**: Xu et al. (2020) concatenated external embeddings *without tampering the original text*. Such embeddings are added alongside the tokenized input and separated using the special *[SEP]* token as follows,

   [CLS] [what] [is] [the] [main] [cause] [of] [hiv] [-] [1] [infection] [in] [children] [?] [SEP] [main] [cause] [hiv 1 infection] [children] [SEP] [context tokens]

4 RESULTS

Table 1 and 2 show the results from our COVID-QA and PubMedQA experiments resp. As our setup was conceptualized as *transductive transfer learning* (same task, different domains) we decided to start with PT-TLMs that had *already learned the problem space* i.e. been FT on a related task (SQuAD w.r.t COVID-QA & SNLI w.r.t PubMedQA). While models FT on SQuAD provided perfect pairing for COVID-QA, we could not find a related model w.r.t the latter since classification tasks seldomly have the same label space. Thus, while we could provide out-of-the-box scores for COVID-QA, we could not for PubMedQA since TLMs FT on SNLI had learned to predict *entailment/contradiction/neutral* whereas we needed them to predict *yes/no/maybe*.

We choose $\text{BERT}_{\text{BASE}}$ and $\text{RoBERTa}_{\text{BASE}}$ as our two general-purpose TLMs and Bio/Sci-BERT for our DS models. As mentioned above, the models were first trained on the SQuAD and SNLI benchmarks before being FT for COVID-QA and PubMedQA and were trained using 5-fold and 10-fold cross-validation (CV) for each dataset respectively. We report the average F1 and EM over all folds for COVID-QA and the average accuracy and F1 on the test set for PubMedQA, for a given epoch across all folds. In other words, for PubMedQA, if a model was trained using 10-fold CV for 10 epochs, and we observed that the best performance on the validation set was on the 4th epoch across all folds, we apply the model from each fold after 4 epochs of training on the test set. Such a method of reporting was chosen on the grounds of providing an unbiased reflection of performance given the size and skewness (majority "yes" answers) of the dataset. For example, if the model from the first fold gave the best validation performance, we felt that using *just* that model on the test set would not be fair for the aforementioned reasons.

In addition to the Mikolov (E-BERT) baseline, we also conduct trials in which the external embeddings were randomized before FT. We felt that such a setup would allow us to gauge how much attention the model is dedicating towards the additional knowledge signals. The intuition was that if the model performance completely crashed using random embeddings, well-homogenized vectors would certainly then be of benefit. On the other hand, if model performance remained relatively the same w.r.t vanilla FT or our approach, we could take it to mean that adding extra embeddings, that were not seen during the PT process, provides no extra advantage.

Finally, regarding concatenation strategies, we observe an overall improvement using DEKCOR concatenation for COVID-QA. This intuitively makes sense because of query distortion resulting from the addition of repeated phrases in the original text. Although, this conjecture seems to be inapplicable to E-BERT, which achieved significant gains on their task or on PubMedQA where gains were seen using the BERTRAM strategy. Thus, we argue that there is merit to both strategies and choosing one might depend on the task at hand. Hence, practitioners need to experiment with both to see which method produces the optimal results.

4.1 COVID-QA RESULTS

Here, utilizing both KGE and DE, integrated via DEKCOR concatenation led to the best performance for the non-DS variants. This approach improves BERT's F1 and EM scores by 1.9% and 6.4% respectively over regular FT, while RoBERTa sees no performance gains in terms of F1, but a 7.6% improvement in EM score w.r.t regular FT.

Comparing our scores for the non-DS models against the E-BERT (Mikolov strategy) baselines we see that BERT achieves 0.6% and 0.4% improvement whereas RoBERTa achieves 1.2% and 3.1% improvement in F1 and EM scores respectively. Despite BERT's vocabulary having only 24 terms in common with our KGE and RoBERTa having 201 terms in common, they still perform relatively well using the Mikolov strategy.

Concerning the DS-models, BioBERT shows a 1% and 0.36% improvement in F1 and EM resp. over vanilla FT whereas SciBERT does not show an increase in F1 but a 1.4% improvement in EM over vanilla FT. Interestingly, the improved EM score for SciBERT arises from regular concatenation.

Overall, we see that by incorporating both KGE and DE, BERT's performance comes very close to BioBERT's and RoBERTa's parallels SciBERT's. In fact, RoBERTa's EM is around 3.1% more than SciBERT. These results show that non-DS models *can* perform at a level of DS-models with additional knowledge infusion.

4.2 PUBMEDQA RESULTS

While results from COVID-QA was found to be more uniform w.r.t concatenation strategy and type of embeddings added i.e. [DEKCOR + avg.(KGE + DE)] = best performance, a similar observation was not seen for PubMedQA. For BERT, we see that KGE & DE along with BERTRAM concatenation led to the best accuracy across all model configurations while the F1 was best for E-BERT and DE + BERTRAM concatenation. Over regular FT, BERTs accuracy improves by 5.2% and F1 by 10%, for the aforementioned configurations. For RoBERTa, we did not see

any performance gains coming from our proposed method. However, in isolation, BERTRAM concatenation of the homogenized KGE yielded the best accuracy (1.6% over regular FT) while E-BERT held the best score in terms of F1 (22.2% above regular FT). It is interesting to see that our best BERT model outperforms the best RoBERTa model in terms of accuracy (0.3%).

Compared to the best E-BERT baseline, BERT sees a 0.26% improvement in accuracy while the F1 stays the same. Our best performing RoBERTa model sees a 5.1% increase in accuracy over E-BERT while F1 drops slightly.

For this task, we see that our external knowledge vectors aren't able to raise the performance of BERT/RoBERTa to DS Bio/Sci-BERT level. We see that BioBERT on its own performs better than with our modifications. On the other hand, SciBERT sees a slight gain in accuracy over regular FT (0.3%) while the F1 remains on par with it.

4.3 Ablation Studies

Following Poerner et al. (2020), we realize that replacing entity tokens with their externally homogenized form would result in poor performance since the semantics of the sentence would get drastically altered rather than enhanced. As such, we only perform concatenation experiments.

We investigate DE and KGE in isolation to understand the influence of each. For COVID-QA, w.r.t vanilla FT of BERT, KGE alone improve F1 and EM performance by 1.5% and 4.7% resp. whereas DE improves it by 1.7% and 6% resp. On the other hand, over vanilla FT for RoBERTa, KGE alone brings down F1 by 2.1% but improves EM by 3.2% and DE alone decreases F1 by 1.7% but increases EM by 2.9%. Thus, we see that for BERT, definition embeddings seem to have more impact than the homogenized KGE while for RoBERTa the situation is split since the KGE improve EM the most while the DE hampers F1 the least. We conjecture that this benefit from DE is due to them resembling TLM vectors the most. The homogenized KGE while providing some benefit, still have to go through a transformation which, we hypothesize, adds a degree of noise.

For PubMedQA, we see that the best KGE model yields a 5.1% and 6.7% increase in accuracy and F1 resp. over regular FT. With just DE we observe a similar improvement in F1 while accuracy improves by 4.8%. For RoBERTa, accuracy and F1 improve over regular FT by 1.6% and 14.8% resp. using only KGE while DE alone from the best model obtains 0.6% and 11.1% improvements resp.

5 Discussion

An interesting takeaway from this study is the performance difference from integrating external knowledge with BERTRAM v/s DEKCOR concatenation. This disparity is more pronounced for RoBERTa than BERT-based methods, which we attribute to the tokenization scheme employed by each model. RoBERTa's tokens may include spaces, so interjecting a new piece of text within a question fundamentally alters how the language is decomposed, and which tokens are presented to the model. BERT's tokens, on the other hand, do not span spaces, so BERT is not as impacted when new tokens are interjected. We attribute the performance disparity in BERT to the corruption of the input that takes place when external knowledge tokens are included. The text is no longer "well-formed" natural language. Additionally, for COVID-QA, we see the most performance improvement in terms of the EM metric. We hypothesize this is because the DS information provided by the external embeddings help pinpoint the DS answers. PubMedQA scores, on the other hand, see overall enhancements for both metrics (F1 being greater) for BERT and RoBERTa based models. This makes sense since as the models get FT, they start understanding the class distribution better which in turn leads to improved F1 scores.

In general, integrating external embeddings provides a performance improvement for non-DS models. Somewhat peculiarly, we see that even adding *random* entity embeddings provides a performance improvement over vanilla FT of DS models. We hypothesize that although the random embeddings do not convey any *real* information, they can serve to denote the presence of different, relevant domain terms. It then holds that the Mikolov approach cannot denote all relevant terms as it requires a vocabulary overlap.

5.1 Method analysis

Under the proposed framework, homogenization need not rely on a vocabulary overlap between the embedding spaces to be aligned. Thus, it allows the method to scale well to domains where there isn't a significant common vocabulary, such as ours. We also see that by integrating external knowledge, non-DS TLMs *can* either perform at a level comparable to DS-TLMs or just show improved performance in general on DS tasks.

Architecture	FT	UMLS Embeds (KGE)	Definition Embeds (DE)	BERTRAM Concat	DEKCOR Concat	F1	EM
$\text{BERT}_{\text{BASE}}$						0.402	0.219
	✓					0.476	0.235
	✓	Random		✓		0.469	0.240
	✓	Random			✓	0.478	0.244
	✓	E-BERT		✓		0.471	0.247
	✓	E-BERT			✓	0.482	0.249
	✓	✓		✓		0.461	0.237
	✓	✓			✓	0.483	0.246
	✓		✓	✓		0.467	0.238
	✓		✓		✓	0.484	0.249
	✓	✓	✓	✓		0.473	0.243
	✓	✓	✓		✓	**0.485**	**0.250**
$\text{RoBERTa}_{\text{BASE}}$						0.437	0.243
	✓					**0.529**	0.278
	✓	Random		✓		0.503	0.275
	✓	Random			✓	0.515	0.281
	✓	E-BERT		✓		0.501	0.275
	✓	E-BERT			✓	0.523	0.290
	✓	✓		✓		0.518	0.287
	✓	✓			✓	0.514	0.276
	✓		✓	✓		0.511	0.277
	✓		✓		✓	0.520	0.286
	✓	✓	✓	✓		0.498	0.273
	✓	✓	✓		✓	**0.529**	**0.299**
BioBERT						0.427	0.238
	✓					0.504	0.275
	✓	✓	✓	✓		0.499	0.269
	✓	✓	✓		✓	**0.509**	**0.276**
SciBERT						0.450	0.248
	✓					**0.537**	0.286
	✓	✓	✓	✓		0.532	**0.290**
	✓	✓	✓		✓	0.531	0.282

Table 1: Average F1/EM over 5-fold CV on CDQA of various model architectures with varying external resources and integration methods.

Analyzing eq. 1, we see that the output of the transformation yields $\hat{y} = f(x^{entity}; W)$ where x^{entity} is the entity embedding to be homogenized. Using a FFNN with the given settings, we have $\hat{y} = W^{(2)}[\tanh(W^{(1)}x^{entity}+b_1)]+b_2$ where $W^{(1)}$ and $W^{(2)}$ are the weight matrices of the linear layers and b_1, b_2 are their respective bias terms. In other words, $\hat{y} = f(x^{entity}; W^{(1)}, W^{(2)})$. This indicates that training the NN, incurs an additional computational overhead since we are trying to learn two parameters instead of one. However, such a cost is mitigated by the fact that most modern-day machines are more than capable of optimizing a simple NN, such as this, without much energy overhead.

Finally, since the homogenization is *specific* to the TLM with which we want to align our embeddings, we must retrain the network for each TLM we want to align with. This factor *might* become a bottleneck if there are a large number of entity embeddings to be homogenized. However, even then, we envisage that the computational cost *would not tend* to be of much concern for the reason mentioned previously.

6 Conclusions and Future Work

In this paper, we investigate the use of external knowledge embedding integration for the task of MRC. Results indicate a degree of promise in the proposed approach, demonstrating how DS information can be added to the input representation of a non-DS model, avoiding the lengthy PT process. However, there is much work yet to be done. First, we

Architecture	FT	UMLS Embeds (KGE)	Definition Embeds (DE)	BERTRAM Concat	DEKCOR Concat	Accuracy	F1
$BERT_{BASE}$	✓					51.28	0.30
	✓	Random		✓		53.66	0.32
	✓	Random			✓	53.72	0.32
	✓	E-BERT		✓		53.66	**0.33**
	✓	E-BERT			✓	53.78	0.32
	✓	✓		✓		53.68	0.32
	✓	✓			✓	53.90	0.32
	✓		✓	✓		53.50	**0.33**
	✓		✓		✓	53.74	0.32
	✓	✓	✓	✓		**53.92**	0.32
	✓	✓	✓		✓	52.98	0.32
$RoBERTa_{BASE}$	✓					52.90	0.27
	✓	Random		✓		52.40	0.32
	✓	Random			✓	52.64	0.29
	✓	E-BERT		✓		50.46	**0.33**
	✓	E-BERT			✓	51.12	0.32
	✓	✓		✓		**53.74**	0.31
	✓	✓			✓	52.92	0.29
	✓		✓	✓		51.04	0.32
	✓		✓		✓	53.24	0.30
	✓	✓	✓	✓		51.88	0.32
	✓	✓	✓		✓	52.36	0.31
BioBERT	✓					**65.06**	**0.45**
	✓	✓	✓	✓		64.76	0.44
	✓	✓	✓		✓	64.42	0.44
SciBERT	✓					63.66	**0.44**
	✓	✓	✓	✓		**63.86**	**0.44**
	✓	✓	✓		✓	63.74	**0.44**

Table 2: Average Accuracy/F1 over 10-fold CV on the test set of PubMedQA, for a given epoch & across all folds, with varying external resources and integration methods.

believe that more research is needed into figuring out alternative strategies for incorporating external embeddings, in the *input representation*. While it might be computationally feasible to train additional knowledge adaptation layers, they still present a processing overhead and increase the overall complexity of the base model. Thus, the research community should expend time exploring this direction.

We recognize that COVID-QA poses two unique challenges viz., the higher average length of contexts and answers and the esoteric domain. As such, we wish to investigate TLMs capable of handling longer texts such as Transformer-XL (Dai et al., 2019) in the future. We particularly avoided doing so in this project since we wanted to see whether lightweight TLMs such as BERT were capable of being improved with minimal modification.

While TLMs such as Bio/Sci-BERT were trained to obtain SOTA performance on DS-tasks, we see a wide gap between their performance on COVID-QA v/s PubMedQA. Scores on the latter task seem to suggest a degree of benefit to PT on DS-texts w.r.t general-purpose TLMs. However, results on the former may make it difficult to consider this a definitive claim. As such, more inspection is needed into the effect of PT on such data.

Finally, further investigation into the overall poor performance of these models is needed. MRC was modelled as a task for mimicking the reading and understanding abilities of children by NLP systems (Hirschman et al., 1999). While TLMs have shown an overall improvement in performance than their predecessors, their performance is still lacking in comparison to their human counterparts. A quick study of the dataset revealed that while the subject matter was quite dense, the answers to the questions and the questions themselves, were straightforward. Thus, we aim to study the different aspects of the MRC process in detail in future works. Answering questions such as whether it is an architecture issue or semantic disconnect will pave the way for true domain generalization.

REFERENCES

Alan R Aronson and François-Michel Lang. An overview of metamap: historical perspective and recent advances. *Journal of the American Medical Informatics Association*, 17(3):229–236, 2010.

Iz Beltagy, Kyle Lo, and Arman Cohan. 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)*, pp. 3615–3620, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1371. URL https://aclanthology.org/D19-1371.

Olivier Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270, 2004.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26, 2013.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. Comet: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4762–4779, 2019.

Zied Bouraoui, Jose Camacho-Collados, and Steven Schockaert. Inducing relational knowledge from bert. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7456–7463, 2020.

Liwei Cai and William Yang Wang. Kbgan: Adversarial learning for knowledge graph embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1470–1480, 2018.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. Enhanced lstm for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1657–1668, 2017.

Pedro Colon-Hernandez, Catherine Havasi, Jason Alonso, Matthew Huggins, and Cynthia Breazeal. Combining pre-trained language models and structured knowledge. *arXiv preprint arXiv:2101.12294*, 2021.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2978–2988, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Daniel Domingo-Fernández, Shounak Baksi, Bruce Schultz, Yojana Gadiya, Reagon Karki, Tamara Raschka, Christian Ebeling, Martin Hofmann-Apitius, and Alpha Tom Kodamullil. Covid-19 knowledge graph: a computable, multi-modal, cause-and-effect knowledge model of covid-19 pathophysiology. *Bioinformatics*, 37(9):1332–1334, 2021.

Lynette Hirschman, Marc Light, Eric Breck, and John D. Burger. Deep read: A reading comprehension system. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*, ACL '99, pp. 325–332, USA, 1999. Association for Computational Linguistics. ISBN 1558606093. doi: 10.3115/1034678.1034731. URL https://doi.org/10.3115/1034678.1034731.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

Qiao Jin, Bhuwan Dhingra, William W Cohen, and Xinghua Lu. Probing biomedical embeddings from language models. *arXiv preprint arXiv:1904.02181*, 2019a.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. Pubmedqa: A dataset for biomedical research question answering. 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)*, pp. 2567–2577, 2019b.

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

Tal Linzen. How can we accelerate progress towards human-like linguistic generalization? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5210–5217, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.465. URL https://aclanthology.org/2020.acl-main.465.

Hanxiao Liu, Zihang Dai, David R So, and Quoc V Le. Pay attention to mlps. *arXiv preprint arXiv:2105.08050*, 2021.

Shanshan Liu, Xin Zhang, Sheng Zhang, Hui Wang, and Weiming Zhang. Neural machine reading comprehension: Methods and trends. *Applied Sciences*, 9(18):3698, 2019.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. K-bert: Enabling language representation with knowledge graph. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34 (03):2901–2908, Apr. 2020. doi: 10.1609/aaai.v34i03.5681. URL https://ojs.aaai.org/index.php/AAAI/article/view/5681.

Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8449–8456, 2020.

Ramon Maldonado, Meliha Yetisgen, and Sanda M Harabagiu. Adversarial learning of knowledge embeddings for the unified medical language system. *AMIA Summits on Translational Science Proceedings*, 2019:543, 2019.

Tomas Mikolov, Quoc V Le, and Ilya Sutskever. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*, 2013.

Timo Möller, Anthony Reina, Raghavan Jayakumar, and Malte Pietsch. Covid-qa: A question answering dataset for covid-19. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, 2020.

Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines. *arXiv preprint arXiv:2006.04884*, 2020.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 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)*, pp. 2463–2473, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1250. URL https://aclanthology.org/D19-1250.

Nina Poerner, Ulli Waltinger, and Hinrich Schütze. E-BERT: Efficient-yet-effective entity embeddings for BERT. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 803–818, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.71. URL https://aclanthology.org/2020.findings-emnlp.71.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training (2018), 2018.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2124. URL https://aclanthology.org/P18-2124.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in bertology: What we know about how bert works. *Transactions of the Association for Computational Linguistics*, 8:842–866, 2020.

Alexey Romanov and Chaitanya Shivade. Lessons from natural language inference in the clinical domain. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 1586–1596, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1187. URL https://www.aclweb.org/anthology/D18-1187.

Timo Schick and Hinrich Schütze. Bertram: Improved word embeddings have big impact on contextualized model performance. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 3996–4007, 2020.

Soumya Sharma, Bishal Santra, Abhik Jana, Santosh Tokala, Niloy Ganguly, and Pawan Goyal. Incorporating domain knowledge into medical NLI using knowledge graphs. 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)*, pp. 6092–6097, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1631. URL https://aclanthology.org/D19-1631.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. Do nlp models know numbers? probing numeracy in embeddings. 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)*, pp. 5307–5315, 2019.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, 2018.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. In *NeurIPS*, 2019.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Doug Burdick, Darrin Eide, Kathryn Funk, Yannis Katsis, Rodney Michael Kinney, Yunyao Li, Ziyang Liu, William Merrill, Paul Mooney, Dewey A. Murdick, Devvret Rishi, Jerry Sheehan, Zhihong Shen, Brandon Stilson, Alex D. Wade, Kuansan Wang, Nancy Xin Ru Wang, Christopher Wilhelm, Boya Xie, Douglas M. Raymond, Daniel S. Weld, Oren Etzioni, and Sebastian Kohlmeier. CORD-19: The COVID-19 open research dataset. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online, July 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.nlpcovid19-acl.1.

Colby Wise, Miguel Romero Calvo, Pariminder Bhatia, Vassilis Ioannidis, George Karypus, George Price, Xiang Song, Ryan Brand, and Ninad Kulkani. Covid-19 knowledge graph: Accelerating information retrieval and discovery for scientific literature. In *Proceedings of Knowledgeable NLP: the First Workshop on Integrating Structured Knowledge and Neural Networks for NLP*, pp. 1–10, 2020.

Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1006–1011, 2015.

Yichong Xu, Chenguang Zhu, Ruochen Xu, Yang Liu, Michael Zeng, and Xuedong Huang. Fusing context into knowledge graph for commonsense reasoning. *arXiv preprint arXiv:2012.04808*, 2020.

Ikuya Yamada, Hiroyuki Shindo, Hideaki Takeda, and Yoshiyasu Takefuji. Joint learning of the embedding of words and entities for named entity disambiguation. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pp. 250–259, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/K16-1025. URL https://aclanthology.org/K16-1025.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*, 2014.

Wenhao Yu, Chenguang Zhu, Yuwei Fang, Donghan Yu, Shuohang Wang, Yichong Xu, Michael Zeng, and Meng Jiang. Dict-bert: Enhancing language model pre-training with dictionary. *arXiv preprint arXiv:2110.06490*, 2021.

Mozhi Zhang, Keyulu Xu, Ken-ichi Kawarabayashi, Stefanie Jegelka, and Jordan Boyd-Graber. Are girls neko or shōjo? cross-lingual alignment of non-isomorphic embeddings with iterative normalization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3180–3189, 2019.

A Appendix

A.1 A Note on Reproducibility

It has been shown previously that FT results from TLMs are not always *stable* i.e. even for the same set of hyperparameters or different random seed values, results may fluctuate (Mosbach et al., 2020). We observed a similar behaviour during FT on our task for all four TLMs. In the spirit of transparency, we report scores from the *first-time-runs* only i.e. we do not report any outliers that had been observed during further trials. However, it should also be said that the scores from successive runs were roughly in the same ballpark. As such, attempting to reproduce these results will yield very *similar*, but perhaps not *exact* scores. Additionally, the scores reported were the **best** among all the parameter variations as we felt that subjecting a model to parameters resulting in lower scores would not be a fair representation of the approach. That being said, we found the following (table 3) hyperparameters produced the best results across all TLMs,

	Dataset	
Hyperparameter	COVID-QA	PubMedQA
Learning Rate	2e-5	1e-5
Epochs	1	10
Batch Size	40	32
Negative Records	4	1

Table 3: Hyperparameters for each task