Entity Extraction in Low Resource Domains with Selective Pre-training of Large Language Models

Aniruddha Mahapatra[1+], **Sharmila Reddy Nangi**[2+]
Aparna Garimella[3], **Anandhavelu Natarajan**[3]
[1]Carnegie Mellon University, USA [2]Stanford University, USA
[3]Adobe Research, India
amahapat@andrew.cmu.edu[1] srnangi@stanford.edu[2]
{garimell, anandvn}@adobe.com[3]

Abstract

Transformer-based language models trained on large natural language corpora have been very useful in downstream entity extraction tasks. However, they often result in poor performances when applied to domains that are different from those they are pretrained on. Continued pretraining using unlabeled data from target domains can help improve the performances of these language models on the downstream tasks. However, using all of the available unlabeled data for pretraining can be time-intensive; also, it can be detrimental to the performance of the downstream tasks, if the unlabeled data is not aligned with the data distribution for the target tasks. Previous works employed external supervision in the form of ontologies for selecting appropriate data samples for pretraining, but external supervision can be quite hard to obtain in low-resource domains. In this paper, we introduce effective ways to select data from unlabeled corpora of target domains for language model pretraining to improve the performances in target entity extraction tasks. Our data selection strategies do not require any external supervision. We conduct extensive experiments for the task of named entity recognition (NER) on seven different domains and show that language models pretrained on target domain unlabeled data obtained using our data selection strategies achieve better performances compared to those using data selection strategies in previous works that use external supervision. We also show that these pretrained language models using our data selection strategies outperform those pretrained on all of the available unlabeled target domain data.

1 Introduction

Named entity recognition (NER) (Lample et al., 2016; Nadeau and Sekine, 2007; Finkel and Manning, 2009) is the task of extracting entities from a given piece of text. NER is useful for several natural language processing (NLP) applications such as information extraction (Chiticariu et al., 2013), retrieval (Banerjee et al., 2019), and language understanding. Over the years, various approaches including rule-based techniques (Farmakiotou et al., 2000), unsupervised learning approaches (Luo et al., 2019), feature-based extractions (Li et al., 2020) have been explored for NER. The recent deep learning-based methods are shown to result in state-of-the-art performances for various domains (Chiu and Nichols, 2016; Peters et al., 2018; Yadav and Bethard, 2018).

Specifically, large pretrained language models such as BERT (Devlin et al., 2019) are widely used for NER, as they mark state-of-the-art performances (Jia et al., 2019). However, these models require large amounts of annotated data which stands as a bottleneck to the training process, particularly when domains having few labeled data are involved. Directly fine-tuning BERT model on small collection of labeled data can yield sub-optimal results. To address this issue, there have been works on pretraining strategies (Liu et al., 2020; Gururangan et al., 2020) to adapt BERT-like language models to target domains by further pretraining them on unlabeled target domain data prior to fine-tuning on labeled data to improve performances in tasks like NER. However, further pretraining language models on all of the available unlabeled data from target domains can be sometimes detrimental to the downstream NER performances, if the entire unlabeled data is not aligned with the entity distribution of the labeled data (Liu et al., 2020).

Specifically, Liu et al. (2020) used two types of unlabeled corpora belonging to the target domain, namely **task-level corpus**, consisting of unlabeled sentences from the target domain that are highly aligned with the distribution of the labeled downstream NER dataset, and **domain-level corpus**, consisting of a very large collection of unlabeled sentences from the target domain, excluding sen-

[+]Work done while authors were at Adobe Research.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 942 - 951
December 7-11, 2022

tences from the task-level corpus. Gururangan et al. (2020) showed the advantages of pretraining on both domain-level and task-level corpora for improving NER performance. However, using the task-level corpus only for further pre-training may be inappropriate, as this is usually very small in size. Additionally, pretraining on the entire domain-level corpus can be challenging, particularly in resource-constrained settings*, and can sometimes produce sub-optimal results due to non-alignment and distribution mismatch. Liu et al. (2020) introduced a strategy to select samples from domain-level corpus to augment the task-level corpus using external supervision in the form of ontologies to improve the downstream NER performance. While this marks the importance of data selection for pretraining using ontology-based filtering, it is constrained by the availability of meta-data and entity information from Wikipedia, and thus, is not easily extensible to other low-resource domains†.

In this paper, we introduce new methods for data selection via *(i)* cosine similarity-based retrieval in embedding space of domain-level corpus generated using pretrained and fine-tuned BERT (fine-tuned on labeled target domain data), *(ii)* entity prediction on domain-level corpus using fine-tuned BERT. Note that both these strategies do not require any additional supervision (*e.g.,* in the form of ontologies) or metadata, thus making them easily extensible to other low-resource domains. Similar to our method, Dai et al. (2019) proposes unsupervised methods of computing similarities between different corpora based on word-embedding and perplexity scores of *bert-base-cased*. However, unlike our method of selecting sentences from the domain-level corpus based on similarity scores, Dai et al. (2019) uses the similarity scores to select the most appropriate source domain (out of multiple source domains) for a particular target domain (in our setting, both domain-level and task-level corpus belong to the same domain). We evaluate our method on i2b2 (Uzuner et al., 2007), Atticus (Hendrycks et al., 2021), and five other domains curated by (Liu et al., 2020) with different amounts of selected sentences for pretraining. Experimental results show that our proposed methods for data selection result in better performances in downstream NER tasks, compared to those proposed by previous works that require external supervision.

*resource-constrained setting(s) refers to settings with limited availability of compute (CPU or GPU) in terms of time and magnitude.

†low-resource domain(s) setting refers to domains with limited or no availability of external metadata

2 Task Definitions

Following the notions in Liu et al. (2020), let $D_C = \{(x_i^D)\}_{i=1}^{N_D}$, and $T_C = \{(x_i^T)\}_{i=1}^{N_T}$ denote the **domain-level** and the **task-level** unlabeled corpora having same domain as $L_C = \{(x_i^L, y_i^L)\}_{i=1}^{N_L}$, which denotes labeled corpus tagged with BIOES for NER, where $L_C \subseteq T_C$, and $N_D \gg N_T \geq N_L$. Given a large unlabeled D_C and relatively small T_C, our task involves selectively sampling sentences $S_C = \{(x_i^S)\}_{i=1}^{N_S}$ from D_C, where $S_C \subset D_C$ and $N_D \gg N_S$, such that pretraining on T_C augmented with S_C increases the performance for NER, without having to pretrain on entire D_C, which can be very time and resource intensive.

For a clearer understanding of D_C, T_C, and L_C we present an **example scenario** involving these 3 different corpus types: Let $\mathcal{X}$ be a small-scaled legal firm that has a limited collection of customer contracts. To ease their workflow, they want to automate the process of extracting named entities (like *contracting-party, contract-amount*, etc.) from contracts using NER models. However, they don't have permission to outsource all the customer contracts to external annotators to create labeled data for NER due to legal restrictions. Out of all the customer contracts, only a very small number (without legal restrictions) can be outsourced for annotations. They also have a large collection of unlabeled SEC contracts (https://www.sec.gov/edgar.shtml) that are publicly available (however, these SEC contracts are somewhat different from customer contracts of $\mathcal{X}$ in terms of format, layout, etc.). In the context of our setting all the customer contracts to form T_C, the small subset of these contracts that can be used for annotations form L_C, and the SEC contracts constitute D_C.

3 Methodology

This section describes the details of our stage-wise method of selecting the pretraining corpus (S_C) from the domain-level corpus (D_C) to pretrain BERT (Devlin et al., 2019), followed by fine-tuning it on a small labeled corpus (L_C) for NER. Figure 1 gives an overview of our data selection strategies for the selective pretraining and the following fine-tuning stages to train BERT for NER.

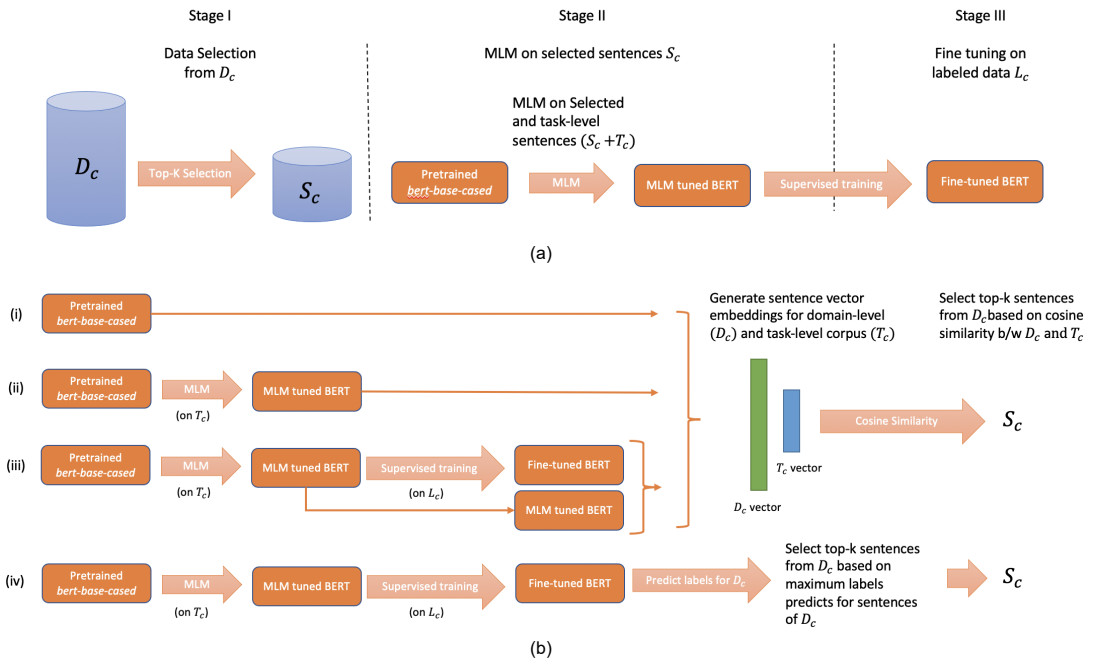


Figure 1: Our method of Selective pre-training and finetuning strategy. (a) describes the overall 3 stage approach. (b) describes all the 4 different strategies used in Stage 1 of (a) for selecting S_C from domain-level corpus (D_C). (i) Cosine Similarity w/ Pretrained BERT, (ii) Cosine Similarity w/ task-level MLM BERT, (iii) Cosine Similarity w/ task-level MLM and NER BERT, and (iv) NER-Selected

3.1 Data Selection Strategies

Continual pretraining on D_C with T_C corpora prior to fine-tuning BERT for NER has shown significant improvement over just fine-tuned BERT w/o task-specific pretraining[††] (Gururangan et al., 2020). However, when size of D_C is extremely large, pretraining on entire corpus can be very time-consuming and resource intensive (see Table 5). Additionally, D_C can contain noisy and unrelated sentences compared to the T_C, leading to reduced effectiveness of domain continual pretraining. Therefore, we devise unsupervised methods to select only the useful sentences from D_C for effective pretraining, without requiring additional domain metadata, such that it can even be applied to any real world low-resource domains.

Aharoni and Goldberg (2020) showed the use of pretrained BERT (Devlin et al., 2019) for effectively clustering sentences consisting of diverse domains. Drawn on this observation, we use the BERT model for selecting sentences from D_C that are more closely associated to T_C based on cosine similarity in sentence embedding space. Moreover, based on results from Liu et al. (2020), sentences in D_C that contain more domain-specialized entities are shown to be more effective for pretraining than randomly selected D_C sentences. Based on these observations, we propose four novel cosine-similarity and NER-Selected based unsupervised methods for data selection from the domain-level corpus for further pretraining (Stage I of figure 1):

- **Cosine Similarity w/ Pretrained BERT**: We use vanilla pretrained *bert-base-cased* model to obtain embeddings of sentences from T_C and D_C as the activations from the last layer of BERT. We compute the mean task-level embedding as the average of all task-level corpus sentence embeddings. We then use it as a query vector to obtain sentences with maximal cosine similarity from the domain-level corpus (D_C).

- **Cosine Similarity w/ task-level MLM BERT**: We first initialize BERT with *bert-base-cased* parameters and then additionally pretrain the BERT model using masked language modeling (details of MLM mentioned in Section 3.2 and 4.1) on the task-level corpus (T_C). This model is referred to as B^1. We use B^1 to select sentences from a domain-level corpus based on the previous method.

[††]in context of this work, w/o task-specific pretraining indicates BERT initialized with *bert-base-cased* parameters, without MLM on additional D_C, T_C or S_C.

- **Cosine Similarity w/ task-level MLM and NER BERT**: We fine-tune B^1 for the NER task (on labeled corpus L_C), termed B^2. Using B^1 and B^2, we obtain task-level and domain-level sentence embeddings. We perform an element-wise concatenation of embeddings obtained from B^1 with the ones obtained using B^2, and then select sentences using these embeddings like the previous method on the aggregated embeddings.

- **NER-Selected**: Instead of using DBpedia Ontology to select sentences with plentiful entities as used in Liu et al. (2020), using B^2, we obtain entity predictions for the sentences in the domain-level corpus. Based on the labels predicted on the domain-level corpus as soft entity labels, we select the sentences with the maximal number of predicted entities. Since B^2 is trained on L_C with domain-specialized entities, it will help in selecting sentences that likely contain similar domain-specialized entities from D_C, thus mitigating the problem of distribution alignment between the NER task data and data used for further pretraining.

3.2 Domain Pretraining and Fine-tuning

Combining selected sentences (S_C) from domain-level corpus (D_C) in Stage I with the task-level corpus (T_C), in 1:1 ratio, we pretrain the BERT model (initialized with *bert-base-cased* parameters) using masked-language modeling (Stage II in figure 1) as specified in Devlin et al. (2019). We mask out 15% of the tokens randomly in the selected sentences and then replace 80% of the masked tokens with a special tag ($[MASK]$), 10% with random, and 10% with original tokens.

Finally, we add a Conditional Random Field (CRF) layer on top of this BERT model and train it for NER on labeled corpus L_C (Stage III of figure 1). More details are specified in Section 4.2.

4 Experiments

4.1 Experimental Setup

For all the experiments, we use BERT model initialized with *bert-base-cased* (Devlin et al., 2019) parameters. The total number of trainable parameter in the BERT model used for experiments are 110M. In all experiments involving pretraining using any corpus type, we pretrain BERT with MLM for 5 epochs with a batch size of 64. We then fine-tune this pretrained BERT on NER task by adding CRF layer on top for 200 epochs (stopping at early convergence on dev set) with learning rate $2e-05$, **L2** regularization of $1e-08$ and AdamW betas $(0.9, 0.999)$. The size of model used for NER task is 442.6 MB. All experiments were performed on Tesla V100 GPU. We evaluate the performance of all the methods with F1 Score (https://github.com/allanj/pytorch_neural_crf). Each experiment is conducted thrice with random seed and the average score is reported.

4.2 Dataset Details

We evaluate our proposed method on seven diverse domain datasets including contracts, medical records, AI, science, politics, music, and literature domains. The Atticus dataset comprises of contracts, i2b2 (Uzuner et al., 2007) contains medical records of patients, and the other five domain datasets are taken from Liu et al. (2020) which contain Wikipedia articles from each of the corresponding domains of AI, science, politics, music, and literature (for example, Science domain contains Wikipedia articles that fall under Science or similar categories).

- **i2b2** (Uzuner et al., 2007): We transform the original dataset of 37.1K labeled sentences into 35.1K domain-level and 2K task-level sentences unlabeled sentences. We use labeled version of the same 2K task-level sentences as L_C for the training of the NER task. We use the original dev and test data provided by Uzuner et al. (2007).

- **CrossNER** (Liu et al., 2020): We directly use the five different domain datasets provided by (Liu et al., 2020), namely AI, Science, Literature, Politics and Music. The dataset is already split into task-level and domain-level corpora, with separate train, test, and dev sets for NER.

- **Atticus** (Hendrycks et al., 2021): Atticus is a recently published Question-Answering dataset for contracts. We only use entity type *contracting-party* from this dataset for NER. We divide all the contracts into splits 70:10:20 for train, dev, and test sets. Out of the 28 different sub-categories of contracts, we select 6 documents from three sub-categories ('Strategic Alliance', 'Development', 'Distributor') as the labeled training set (L_C) and all

Domain	Unlabeled Corpus	Labeled Corpus L_c			Entity Categories
	Domain-level D_c	Train	Dev	Test	
i2b2	35.1K	2K	3.9K	35.5K	date, patient, doctor, medical-record, age, hospital, phone, idnum, username, street, city, state, zip, device, profession, country, organization, location-other, fax, email
AI	111K	100	350	431	field, task, product, algorithm, researcher, metrics, university, country, person, organization, location, miscellaneous
Science	2.08M	100	350	431	scientist, person, university, organization, country, location, discipline, enzyme, protein, chemical compound, chemical element, event, astronomical object, academic journal, award, theory, miscellaneous
Politics	3.39M	200	450	450	politician, person, organization, political party, event, election, country, location, miscellaneous
Music	4.46M	100	380	456	Music genre, song, band, album, Musical artist, Musical instrument, award, event, country, location, organization, person, miscellaneous
Literature	3.32M	100	400	416	book, writer, award, poem, event, magazine, person, location, organization, country, miscellaneous
Atticus	38K	1.9K	8K	16.6K	contracting-party

Table 1: Dataset details of the different domains used for experimentation. Table describes the number of sentences in unlabeled domain-level corpus, train, test, and dev set of labeled corpus, and entity types of each domain.

the documents under these 3 sub-categories as task-level sentences (T_C) from the 70% total train split. The rest of the documents from the total train split (from the remaining 25 sub-categories) that are not included in the task-level corpus are considered from the domain-level corpus (D_C).

Note that we modify the original i2b2 and Atticus datasets to our problem setting (using the method described above). More details on the number of domain-level D_C, task-level T_C, selected S_C sentences and labeled train, test and dev splits along with entity types are provided in Tables 1 and 2. For fairness of comparison of our methods with Entity-level data selection strategy (Section 4.3) used in Liu et al. (2020), we use the same number of selected sentences (S_C) for each of the Liu et al. (2020) datasets (as mentioned for respective domains in Entity-level corpus of (Liu et al., 2020)) for experimentation. For i2b2 and Atticus datasets, we take the number of sentences in S_C to be approximately $\frac{1}{2}$ that of D_C.

4.3 Baseline Methods

We compare the effectiveness of our data-selection and pre-training strategies against the following baselines:

- **w/o task-specific pretraining**[††]: We use a *bert-base-cased* model (Devlin et al., 2019) and fine-tune it for the NER task on L_C.
- **Task-level pertaining (TAPT)**: We initialize the BERT model with *bert-base-cased* parameters and pretrain on the task-level corpus with MLM, then, fine-tune for the NER task (analogous to TAPT in Gururangan et al. (2020)).
- **Domain-level pretraining (DAPT)**: We first initialize the BERT model with *bert-base-cased* parameters and pretrain on the domain-level corpus with MLM, followed by fine-tuning for the NER task (analogous to DAPT in Gururangan et al. (2020)).
- **Task w/ Domain-level pretraining (TAPT + DAPT)**: We combine the domain-level and task-level corpora in 1:1 ratio. We initialize BERT with *bert-base-cased* parameters and pretrain on the combined corpus with MLM, then, fine-tune for the NER task (analogous to TAPT + DAPT in Gururangan et al. (2020)).
- **Entity-level w/ Task-level pretraining**: We use the entity-level corpus, provided by Liu et al. (2020), that comprises of sentences from domain-level corpus having plentiful entities which are obtained by leveraging knowledge from DBpedia Ontology. We combine the entity-level and task-level corpora in 1:1 ratio for pretraining (integrated corpus mentioned in Liu et al. (2020)). Since we use token-level MLM in all our methods, for fair comparison of the effectiveness of our selected sentences that of Liu et al. (2020), we compare with their method which performs token-level pretraining with integrated corpus.
- **Perplexity Score (PPL)**: Dai et al. (2019) uses language model perplexity score to select the most appropriate source domain, from a

	i2b2	AI	Science	Politics	Music	Litera.	Atticus
Domain-level D_c	35.1K (1x)	111K (1x)	2.08M (1x)	3.39M (1x)	4.46M (1x)	3.32M (1x)	38K (1x)
Task-level T_c	2K (0.05x)	7.4K (0.06x)	124K (0.05x)	501K (0.14x)	826K (0.18x)	429K (0.12x)	8.4K (0.22x)
Selected S_c	16K (0.45x)	38.4K (0.34x)	396K (0.19x)	692K (0.20x)	1.19M (0.26x)	1.01M (0.30x)	20K (0.52x)

Table 2: Number of sentences in different corpus types (D_C, T_C and S_C) for each domain. The number in the brackets represents the size ratio between the corresponding corpus type and the domain-level corpus (D_C).

Models	Corpus/ Selection Method	i2b2	AI	Science	Politics	Music	Litera.	Atticus
BERT-based	Cosine-Similarity w/ pretrained BERT	72.84	56.54	67.24	72.36	72.85	**64.77**	64.23
	Cosine-Similarity w/ MLM tuned BERT	**74.22**	56.19	67.22	72.44	72.95	64.49	64.53
	Cosine-Similarity w/ MLM and NER tuned BERT	73.15	56.42	67.13	72.65	72.39	64.74	**66.03**
	NER-Selected	74.15	**56.92**	**67.54**	**72.69**	**73.55**	64.29	63.51
Baseline Methods								
BERT-based	w/o task-specific pretraining	70.61	53.35	63.86	69.7	67.13	61.35	61.76
	Task-level pretraining (TAPT)	72.24	55.08	66.02	70.43	70.61	62.33	63.41
	Domain-level pretraining (DAPT)	73.25	55.38	66.27	72.23	71.35	63.2	63.22
	Task w/ Domain-level pretraining (TAPT+DAPT)	72.49	55.52	66.56	72.62	71.58	64.57	63.74
CrossNER	Entity-level w/ Task-level pretraining	-	56.44	67.15	72.43	72.42	64.36	-
Dai et al. (2019) modified	Perplexity Score (PPL)	72.95	56.71	66.4	72.15	73.11	64.25	64.81

Table 3: F1 score comparison of our and different baselines methods on all the domain datasets. Our model outperforms all the baselines methods across all domains. The F1 scores are averaged over 3 experimental runs. The F1 scores for CrossNER (Liu et al., 2020) method on i2b2 and Atticus domains are absent as this method of data selection requires external metadata (DBpedia Ontology) which is not present for both these domains.

Method	Entity Types									
	Book	Writer	Award	Event	Magazine	Person	Location	Organization	Country	Misc.
Domain-level pretraining (DAPT)	69.2	80.49	85.7	62.78	71.55	21.03	49.55	62.61	67.31	36.71
Cosine-Similarity w/ MLM and NER tuned BERT	**71.61**	**81.09**	**86.62**	**65.06**	**75.47**	**21.34**	**50**	**62.86**	**68.29**	**36.74**

Table 4: F1-scores for the different entity-types in the Literature domain. Scores are reported for 2 different methods, 'Domain-level pretraining (DAPT)' and 'Cosine Similarity w/ MLM and NER tuned BERT'. Scores are averaged over 3 experimental runs.

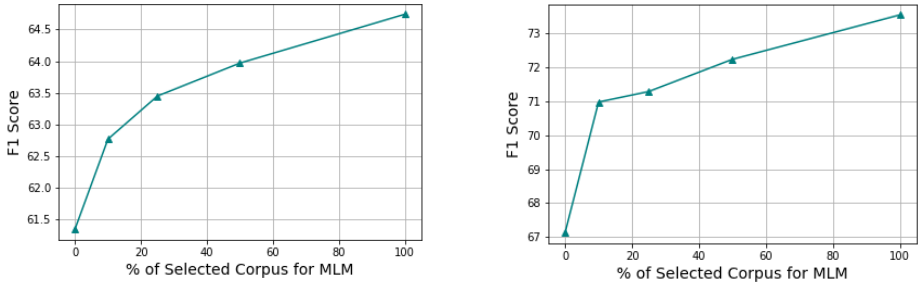
collection of different domains for a given target domain. Specifically, they train a model on target corpus and compute perplexity score of each source corpora using this model, and then select the source that gives the lowest perplexity score. Instead, we modify this method for our scenario by first training a BERT model on the task-level corpus and using it to select sentences from domain-level corpus having minimum perplexity [‡].

5 Results and Analysis

5.1 Performance v/s Data Selection Strategies

From the results of our experiments in Table 3, we see that pretraining using selected sentences using our methods outperform all baselines for all of the domain datasets. Figure 3 (b) shows qualitative examples of NER predictions from the Atticus test set, using our method: 'Cosine-Similarity w/ MLM and NER tuned BERT' and 'w/o task-specific pretraining'. Our method is able to predict 'contacting-party' much more accurately than the 'w/o task-specific pretraining' model. According to Table 2, although the size of the selected corpus S_C is around half or less than half that of the domain-level corpus, the pretraining using S_C improves NER performance than pretraining on D_C or D_C and T_C combined. Based on this it can be hypothesized that selection of sentences using cosine similarity over BERT embeddings filter many noisy sentences that might not be very related to the task-domain corpus as only the top-most sentences are selected that lie close to mean task-level corpus embedding. Additionally, NER-Selected method removes sentences that might not contain domain-specific entities related to the particular NER task, increasing the effectiveness of pretraining (See fig-

[‡]Note that we do not use the 'Target Vocabulary Covered' (TVC) or 'Word Vector Variance' (WVV) as in Dai et al. (2019). TVC would not make much meaning to calculate on a sentence level (unlike for Dai et al. (2019) where they use TVC across entire domain corpora). Similarly, WVV is used to calculate word vector variance across pretranined word vectors on 2 different domains (and not across sentences) and can't be modified for our scenario of selecting sentences



(a) Literature
(Cosine Similarity w/ MLM and NER Tuned BERT)

(b) Music
(NER-Selected)

Figure 2: Comparison of F1-scores for different amounts of selected corpus (S_C) (w/ task-level corpus (T_C)). Scores are averaged over 3 experimental runs.

Rejected Sentences
1. Save by strange names uncouth to English ears. 2. Excessive or inappropriate links will be removed. 3. 171,000 sailors.
Selected Sentences
1. Surface Alberto is defined as the ratio of Radiosity (radiometry) | radiosity to the irradiance. 2. Aristotle's writings on motion remained influential until the Early Modern period. 3. Archimedes corrected Aristotle's theory that bodies move towards their natural resting places ; metal boats can float if they Archimedes ' principal | displace enough water ; floating depends in Archimedes ' scheme on the mass and volume of the object , not as Aristotle thought its elementary composition.

(a)

Berkeley Lights, Inc, 5858 Horton Street, Suit 320 Emeryville, CA 94608 Attn: [***].

Berkeley Lights, Inc, 5858 Horton Street, Suit 320 Emeryville, CA 94608 Attn: [***].

Berkeley Lights, Inc, 5858 Horton Street, Suit 320 Emeryville, CA 94608 Attn: [***].

If to Sutro: Sutro Biopharma, Inc. 310 Utah Ave, Suite 150 South San Francisco CA 94080 Attention: Chief Executive Officer

If to Sutro: Sutro Biopharma, Inc. 310 Utah Ave, Suite 150 South San Francisco CA 94080 Attention: Chief Executive Officer

If to Sutro: Sutro Biopharma, Inc. 310 Utah Ave, Suite 150 South San Francisco CA 94080 Attention: Chief Executive Officer

(b)

Figure 3: Figure shows (a) examples of sentences selected and filtered out by the NER-Selected method for the Science domain, (b) qualitative examples of contracting-party entity predicted by our method 'Cosine-Similarity w/ MLM and NER tuned BERT' v/s 'w/o task-specific pretraining' on two Atticus dataset samples. [Pink] indicates ground-truth entity mentions, [Blue] indicates predictions by our method 'Cosine-Similarity w/ MLM and NER tuned BERT', [Yellow] indicates precitions by 'w/o task-specific pretraining' method.

ure 3 (a)). Furthermore, integrating the selected corpus and task-level corpus is able to consistently boost the downstream NER performance compared to utilizing other corpus types although the size of this corpus is still smaller than the domain-level corpus. This is because it ensures that the pretraining corpus contains content that is explicitly related to the NER task in the target task-level corpus while also making it relatively larger than the task-level corpus itself. The results suggest that the corpus content is essential for the effectiveness of continual pretraining. Surprisingly, this pretraining strategy is still effective for the i2b2, Atticus, and AI domains even though the corpus sizes in these domains are relatively small, which illustrates the effectiveness of our method in settings with less availability of unlabeled domain-level corpus.

5.2 Performance v/s Pretraining Corpus Size

Since such large domain-level corpora might not be always available in real-world scenarios, we investigate the performance of pretraining on the different quantities of selected sentences (S_C). From figure 2, it is evident that as the number of selected sentences increases, the performance keeps improving (in this figure it is shown for both Literature and Music domains). This implies that a larger corpus size is better for pretraining. However, the slope of the performance gain curve becomes less when we increase the number of sentences for MLM beyond 50% of S_C. Our method of pretraining can also be used in resource-constrained settings, where it is infeasible to pretrain with MLM on extremely large corpora. Table 5 shows the performance with the time required for MLM pretraining on different cor-

Corpus	Literature			Music		
	# sentences	# F1 Score	Time for MLM (hh:mm:ss)	# sentences	# F1 Score	Time for MLM (hh:mm:ss)
None	0	61.35	0	0	67.13	0
T_C	429K	62.33	6:24:23	826K	70.61	11:10:30
$T_C + S_C$ (10%)	439K	62.77	6:33:26	838K	70.98	11:20:13
$T_C + S_C$ (25%)	682K	63.45	10:11:01	1.12M	71.29	15:12:55
$T_C + S_C$ (50%)	935K	63.97	13:57:40	1.42M	72.24	21:37:12
$T_C + S_C$ (100%)	1.44M	**64.74**	21:31:01	2.02M	**73.55**	30:41:35
D_C	3.32M	63.2	48:14:33	4.46M	71.35	62:14:29
$D_C + T_C$	3.75M	64.57	56:02:13	5.29M	71.58	80:24:54

Table 5: Comparison of performance (F1 score) and time (in hours:minutes:seconds) for Literature and Music domains using different amounts and corpora (T_C, S_C, and D_C) of sentences used for MLM pretraining. We observe that for both domains, T_C + S_C achieves the best performance, with almost half the number of sentences used and less than 2.5x the time required for pretraining compared to T_C + D_C, showing the efficiency of our method over naive pretraining on entire corpora. The F1 scores are reported over 3 experimental runs. Results for Literature domain are calculated using 'Cosine-Similarity w/ MLM and NER tuned BERT' and Music domain using 'NER Selected' methods. We show 2 different domains with different methods of sentence selection to demonstrate the robustness for our different selection methods in terms of both time and F1 score improvement.

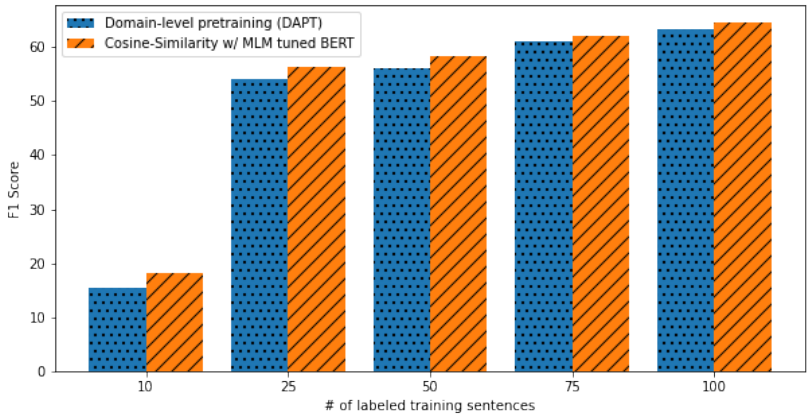


Figure 4: Comparison of F1-scores for different amounts of training labeled corpus (L_C) on Literature domain. Scores are reported for 2 different methods, 'Domain-level pretraining (DAPT)' and 'Cosine Similarity w/ MLM and NER tuned BERT'. Scores are averaged over 3 experimental runs.

pus sizes and types for two different domains with two different methods of data selection. For the Literature domain with 'Cosine-Similarity w/ MLM and NER tuned BERT', we see that we achieve an F1 of 64.67 when using T_C with S_C compared to pretraining using entire D_C along with T_C (F1 64.57), even though it takes less than 2.5 times the time required for MLM pretraining with almost half the number of total sentences. A similar trend can be observed for the Music domain.

5.3 Fine-grained Comparison

In this section, we explore the effectiveness of our pretraining strategy using selected sentences on individual entities belonging to the same domain over 'Domain-level pretraining (DAPT)' method. Table 4 shows the performance of the majority of the entities for the Literature domain using two methods 'Domain-level pretraining (DAPT)' and 'Cosine-Similarity w/ MLM and NER tuned BERT'. We see that the performance has increased for all entity types when pretraining on selected sentences using our method has been used. We observe that the performance on the *person* entity type is comparatively lower than all other entity types for both methods. It may be due to the hierarchical category structure of this domain that causes the model to get confused between *person* and *writer* entity types. It can be also seen that the performance of domain-specific entities (like 'book', 'writer', 'event' and 'magazine') has improved much more than the improvement of the non domain-specific generic en-

tity types (like 'person', 'location', 'organization') when our method of 'Cosine-Similarity w/ MLM and NER tuned BERT' is used over 'Domain-level pretraining (DAPT)' for pretraining.

5.4 Performance v/s Labeled Corpus Size

From figure 4, it can be inferred that the performance decreases drastically as the number of labeled samples in L_C used for fine-tuning BERT-CRF is reduced. Specifically, the performance becomes extremely low for Literature when there are only 10 labeled samples (18.3 F1 for 'Cosine-Similarity w/ MLM tuned BERT' and 15.45 F1 for 'Domain-level pretraining (DAPT)'). Although the performance of our method is always better than 'Domain-level pretraining (DAPT)' when the number of labeled samples are extremely low (10), the improvement from our method over 'Domain-level pretraining (DAPT)' is significant ($\sim$18% increase), and this margin of improvement gets smaller as we increase the number of labeled sentences ($\sim$2% at 100 labeled samples). This can be explained by the fact that the BERT model is able to gain domain knowledge from the pretraining data and thus has the ability to better predict entity labels.

6 Conclusion

We propose novel methods of data selection techniques, which do not require additional external supervision, for pretraining BERT for NER. In addition to being illustrated on 7 diverse domains, our method can be easily extended to NER for any new domain with very scarce labeled data and plenty of domain unlabeled data. Experiments on multiple domain datasets demonstrate that our selection techniques are better than naive pretraining on the entire domain corpus and also achieve better performance compared to state-of-the-art data selection methods using external knowledge bases. Since we may not get domain labels for every corpora, in future we will try to extend our work with corpus consisting of a mixture of different domains.

7 Limitations

Our method works well when both the task-level and domain-level corpus to belong to the same domain or in similar domain, i.e, if the task-level corpus belongs to the science domain, then the domain-level corpus should also belong to science/similar domain. However, since clear domain labels may not be present for all corpora in the wild, one may have to choose an appropriate domain-level corpus corresponding to the task-level corpus. The second limitation of this work is that we assume the domain-level corpus consists only of a single domain, though, a real-world corpus might consist of mixture of sentences belonging to different domains. In this work, we have not verified how effective our selection strategies will be in selecting sentences for pretraining in such a scenario where domain-level corpus consists of a mixture of domains (with domains present other than the domain label of task-level corpus). Additionally, all our data selection strategies, except the 'Cosine-Similarity w/ pretrained BERT' use domain-specific models of data selection, hence there can not be a unified model that can be applied for data selection for any domain without additional pretraining.

References

Roee Aharoni and Yoav Goldberg. 2020. Unsupervised domain clusters in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Partha Sarathy Banerjee, Baisakhi Chakraborty, Deepak Tripathi, Hardik Gupta, and Sourabh S Kumar. 2019. A information retrieval based on question and answering and ner for unstructured information without using sql. *Wireless Personal Communications*, 108(3):1909–1931.

Laura Chiticariu, Yunyao Li, and Frederick Reiss. 2013. Rule-based information extraction is dead! long live rule-based information extraction systems! In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 827–832.

Jason P.C. Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Xiang Dai, Sarvnaz Karimi, Ben Hachey, and Cecile Paris. 2019. Using similarity measures to select pretraining data for ner. *arXiv preprint arXiv:1904.00585*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dimitra Farmakiotou, Vangelis Karkaletsis, John Koutsias, George Sigletos, Constantine D Spyropoulos, and Panagiotis Stamatopoulos. 2000. Rule-based named entity recognition for greek financial texts. In *Proceedings of the Workshop on Computational lexicography and Multimedia Dictionaries (COMLEX 2000)*, pages 75–78. Citeseer.

Jenny Rose Finkel and Christopher D Manning. 2009. Nested named entity recognition. In *Proceedings of the 2009 conference on empirical methods in natural language processing*, pages 141–150.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of ACL*.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. Cuad: An expert-annotated nlp dataset for legal contract review. *NeurIPS*.

Chen Jia, Xiaobo Liang, and Yue Zhang. 2019. Cross-domain NER using cross-domain language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2464–2474, Florence, Italy. Association for Computational Linguistics.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2020. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2020. Crossner: Evaluating cross-domain named entity recognition.

Ying Luo, Hai Zhao, and Junlang Zhan. 2019. Named entity recognition only from word embeddings. *arXiv preprint arXiv:1909.00164*.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Özlem Uzuner, Yuan Luo, and Peter Szolovits. 2007. Evaluating the state-of-the-art in automatic de-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563.

Vikas Yadav and Steven Bethard. 2018. A survey on recent advances in named entity recognition from deep learning models. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2145–2158, Santa Fe, New Mexico, USA. Association for Computational Linguistics.