Self-Alignment Pretraining for Biomedical Entity Representations

Fangyu Liu♣, Ehsan Shareghi◇,♣, Zaiqiao Meng♣, Marco Basaldella♡∗, Nigel Collier♣

♣Language Technology Lab, TAL, University of Cambridge
◇Department of Data Science & AI, Monash University ♡Amazon Alexa
♣{fl399, zm324, nhc30}@cam.ac.uk
◇ehsan.shareghi@monash.edu ♡mbbasald@amazon.co.uk

Abstract

Despite the widespread success of self-supervised learning via masked language models (MLM), accurately capturing fine-grained semantic relationships in the biomedical domain remains a challenge. This is of paramount importance for entity-level tasks such as entity linking where the ability to model entity relations (especially synonymy) is pivotal. To address this challenge, we propose SAPBERT, a pretraining scheme that self-aligns the representation space of biomedical entities. We design a scalable metric learning framework that can leverage UMLS, a massive collection of biomedical ontologies with 4M+ concepts. In contrast with previous pipeline-based hybrid systems, SAPBERT offers an elegant one-model-for-all solution to the problem of medical entity linking (MEL), achieving a new state-of-the-art (SOTA) on six MEL benchmarking datasets. In the scientific domain, we achieve SOTA even without task-specific supervision. With substantial improvement over various domain-specific pretrained MLMs such as BIOBERT, SCIBERT and PUB-MEDBERT, our pretraining scheme proves to be both effective and robust.[1]

Figure 1: The t-SNE (Maaten and Hinton, 2008) visualisation of UMLS entities under PUBMEDBERT (BERT pretrained on PubMed papers) & PUBMED-BERT+SAPBERT (PUBMEDBERT further pretrained on UMLS synonyms). The biomedical names of different concepts are hard to separate in the heterogeneous embedding space (left). After the self-alignment pretraining, the same concept's entity names are drawn closer to form compact clusters (right).

1 Introduction

Biomedical entity[2] representation is the foundation for a plethora of text mining systems in the medical domain, facilitating applications such as literature search (Lee et al., 2016), clinical decision making (Roberts et al., 2015) and relational knowledge discovery (e.g. chemical-disease, drug-drug and protein-protein relations, Wang et al. 2018). The heterogeneous naming of biomedical concepts poses a major challenge to representation learning. For instance, the medication *Hydroxychloroquine* is often referred to as *Oxichlorochine* (alternative name), *HCQ* (in social media) and *Plaquenil* (brand name).

MEL addresses this problem by framing it as a task of mapping entity mentions to unified concepts in a medical knowledge graph.[3] The main bottleneck of MEL is the quality of the entity representations (Basaldella et al., 2020). Prior works in this domain have adopted very sophisticated text pre-processing heuristics (D'Souza and Ng, 2015; Kim et al., 2019; Ji et al., 2020; Sung et al., 2020) which can hardly cover all the variations of biomedical names. In parallel, self-supervised learning has shown tremendous success in NLP via leveraging the masked language modelling (MLM)

---

[1]For code and pretrained models, please visit: https://github.com/cambridgeltl/sapbert.

[2]In this work, *biomedical entity* refers to the surface forms of biomedical concepts, which can be a single word (e.g. *fever*), a compound (e.g. *sars-cov-2*) or a short phrase (e.g. *abnormal retinal vascular development*).
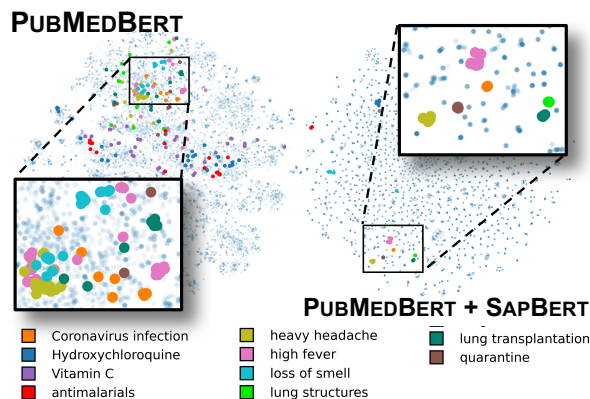
[3]Note that we consider only the biomedical entities themselves and not their contexts, also known as medical concept normalisation/disambiguation in the BioNLP community.

objective to learn semantics from distributional representations (Devlin et al., 2019; Liu et al., 2019). Domain-specific pretraining on biomedical corpora (e.g. BIOBERT, Lee et al. 2020 and BIOMEGATRON, Shin et al. 2020) have made much progress in biomedical text mining tasks. Nonetheless, representing medical entities with the existing SOTA pretrained MLMs (e.g. PUBMEDBERT, Gu et al. 2020) as suggested in Fig. 1 (left) does not lead to a well-separated representation space.

To address the aforementioned issue, we propose to pretrain a Transformer-based language model on the biomedical knowledge graph of UMLS (Bodenreider, 2004), the largest interlingua of biomedical ontologies. UMLS contains a comprehensive collection of biomedical synonyms in various forms (UMLS 2020AA has 4M+ concepts and 10M+ synonyms which stem from over 150 controlled vocabularies including MeSH, SNOMED CT, RxNorm, Gene Ontology and OMIM).[4] We design a self-alignment objective that clusters synonyms of the same concept. To cope with the immense size of UMLS, we sample hard training pairs from the knowledge base and use a scalable metric learning loss. We name our model as **S**elf-**a**ligning **p**retrained **BERT** (SAPBERT).

Being both simple and powerful, SAPBERT obtains new SOTA performances across all six MEL benchmark datasets. In contrast with the current systems which adopt complex pipelines and hybrid components (Xu et al., 2020; Ji et al., 2020; Sung et al., 2020), SAPBERT applies a much simpler training procedure without requiring any pre- or post-processing steps. At test time, a simple nearest neighbour's search is sufficient for making a prediction. When compared with other domain-specific pretrained language models (e.g. BIOBERT and SCIBERT), SAPBERT also brings substantial improvement by up to 20% on accuracy across all tasks. The effectiveness of the pretraining in SAPBERT is especially highlighted in the scientific language domain where SAPBERT outperforms previous SOTA even without fine-tuning on any MEL datasets. We also provide insights on pretraining's impact across domains and explore pretraining with fewer model parameters by using a recently introduced ADAPTER module in our training scheme.
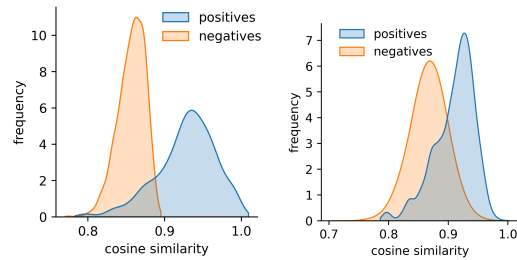

Figure 2: The distribution of similarity scores for all sampled PUBMEDBERT representations in a mini-batch. The left graph shows the distribution of **+** and **-** pairs which are easy and already well-separated. The right graph illustrates larger overlap between the two groups generated by the online mining step, making them harder and more informative for learning.

2 Method: Self-Alignment Pretraining

We design a metric learning framework that learns to self-align synonymous biomedical entities. The framework can be used as both pretraining on UMLS, and fine-tuning on task-specific datasets. We use an existing BERT model as our starting point. In the following, we introduce the key components of our framework.

Formal Definition. Let $(x, y) \in \mathcal{X} \times \mathcal{Y}$ denote a tuple of a name and its categorical label. For the self-alignment pretraining step, $\mathcal{X} \times \mathcal{Y}$ is the set of all (name, CUI[5]) pairs in UMLS, e.g. (*Remdesivir*, C4726677); while for the fine-tuning step, it is formed as an entity mention and its corresponding mapping from the ontology, e.g. (*scratchy throat*, 102618009). Given any pair of tuples $(x_i, y_i), (x_j, y_j) \in \mathcal{X} \times \mathcal{Y}$, the goal of the self-alignment is to learn a function $f(\cdot; \theta) : \mathcal{X} \to \mathbb{R}^d$ parameterised by θ. Then, the similarity $\langle f(x_i), f(x_j) \rangle$ (in this work we use cosine similarity) can be used to estimate the resemblance of x_i and x_j (i.e., high if x_i, x_j are synonyms and low otherwise). We model f by a BERT model with its output [CLS] token regarded as the representation of the input.[6] During the learning, a sampling procedure selects the informative pairs of training samples and uses them in the pairwise metric learning loss function (introduced shortly).

Online Hard Pairs Mining. We use an online hard triplet mining condition to find the most

---

[5]In UMLS, CUI is the **C**oncept **U**nique **I**dentifier.

[6]We tried multiple strategies including first-token, mean-pooling, [CLS] and also NOSPEC (recommended by Vulić et al. 2020) but found no consistent best strategy (optimal strategy varies on different *BERTs).

informative training examples (i.e. hard positive/negative pairs) within a mini-batch for efficient training, Fig. 2. For biomedical entities, this step can be particularly useful as most examples can be easily classified while a small set of very hard ones cause the most challenge to representation learning.[7] We start from constructing all possible triplets for all names within the mini-batch where each triplet is in the form of (x_a, x_p, x_n). Here x_a is called *anchor*, an arbitrary name in the mini-batch; x_p a positive match of x_a (i.e. $y_a = y_p$) and x_n a negative match of x_a (i.e. $y_a \neq y_n$). Among the constructed triplets, we select out all triplets that violate the following condition:

$$\|f(x_a) - f(x_p)\|_2 < \|f(x_a) - f(x_n)\|_2 + \lambda, \quad (1)$$

where λ is a pre-set margin. In other words, we only consider triplets with the negative sample closer to the positive sample by a margin of λ. These are the hard triplets as their original representations were very far from correct. Every hard triplet contributes one hard positive pair (x_a, x_p) and one hard negative pair (x_a, x_n). We collect all such positive & negative pairs and denote them as $\mathcal{P}, \mathcal{N}$. A similar but not identical triplet mining condition was used by Schroff et al. (2015) for face recognition to select hard negative samples. Switching-off this mining process, causes a drastic performance drop (see Tab. 2).

Loss Function. We compute the pairwise cosine similarity of all the BERT-produced name representations and obtain a similarity matrix $\mathbf{S} \in \mathbb{R}^{|\mathcal{X}_b| \times |\mathcal{X}_b|}$ where each entry $\mathbf{S}_{ij}$ corresponds to the cosine similarity between the i-th and j-th names in the mini-batch b. We adapted the Multi-Similarity loss (MS loss, Wang et al. 2019), a SOTA metric learning objective on visual recognition, for learning from the positive and negative pairs:

$$\mathcal{L} = \frac{1}{|\mathcal{X}_b|} \sum_{i=1}^{|\mathcal{X}_b|} \left(\frac{1}{\alpha} \log \left(1 + \sum_{n \in \mathcal{N}_i} e^{\alpha(\mathbf{S}_{in} - \epsilon)} \right) \right.$$
$$\left. + \frac{1}{\beta} \log \left(1 + \sum_{p \in \mathcal{P}_i} e^{-\beta(\mathbf{S}_{ip} - \epsilon)} \right) \right), \quad (2)$$

where α, β are temperature scales; ϵ is an offset applied on the similarity matrix; $\mathcal{P}_i, \mathcal{N}_i$ are indices

of positive and negative samples of the *anchor* i.[8]

While the first term in Eq. 2 pushes negative pairs away from each other, the second term pulls positive pairs together. This dynamic allows for a re-calibration of the alignment space using the semantic biases of synonymy relations. The MS loss leverages similarities among and between positive and negative pairs to re-weight the importance of the samples. The most informative pairs will receive more gradient signals during training and thus can better use the information stored in data.

3 Experiments and Discussions

3.1 Experimental Setups

Data Preparation Details for UMLS Pretraining. We download the full release of UMLS 2020AA version.[9] We then extract all English entries from the MRCONSO.RFF raw file and convert all entity names into lowercase (duplicates are removed). Besides synonyms defined in MRCONSO.RFF, we also include tradenames of drugs as synonyms (extracted from MRREL.RRF). After pre-processing, a list of 9,712,959 (name, CUI) entries is obtained. However, random batching on this list can lead to very few (if not none) positive pairs within a mini-batch. To ensure sufficient positives present in each mini-batch, we generate offline positive pairs in the format of (name$_1$, name$_2$, CUI) where name$_1$ and name$_2$ have the same CUI label. This can be achieved by enumerating all possible combinations of synonym pairs with common CUIs. For balanced training, any concepts with more than 50 positive pairs are randomly trimmed to 50 pairs. In the end we obtain a training list with 11,792,953 pairwise entries.

UMLS Pretraining Details. During training, we use AdamW (Loshchilov and Hutter, 2018) with a learning rate of 2e-5 and weight decay rate of 1e-2. Models are trained on the prepared pairwise UMLS data for 1 epoch (approximately 50k iterations) with a batch size of 512 (i.e., 256 pairs per mini-batch). We train with Automatic Mixed Precision (AMP)[10] provided in PyTorch 1.7.0. This takes approximately 5 hours on our machine (con-

---

[7]Most of *Hydroxychloroquine*'s variants are easy: *Hydroxychlorochin, Hydroxychloroquine (substance), Hidroxicloroquina*, but a few can be very hard: *Plaquenil* and *HCQ*.

[8]We explored several loss functions such as InfoNCE (Oord et al., 2018), NCA loss (Goldberger et al., 2005), simple cosine loss (Phan et al., 2019), max-margin triplet loss (Basaldella et al., 2020) but found our choice is empirically better. See App. §B.2 for comparison.

[9]https://download.nlm.nih.gov/umls/kss/2020AA/umls-2020AA-full.zip

[10]https://pytorch.org/docs/stable/amp.html

model	scientific language								social media language			
	NCBI		BC5CDR-d		BC5CDR-c		MedMentions		AskAPatient		COMETA	
	@1	@5	@1	@5	@1	@5	@1	@5	@1	@5	@1	@5
vanilla BERT (Devlin et al., 2019)	67.6	77.0	81.4	89.1	79.8	91.2	39.6	60.2	38.2	43.3	40.4	47.7
+ SapBert	91.6	95.2	92.7	95.4	96.1	98.0	52.5	72.6	68.4	87.6	59.5	76.8
BioBert (Lee et al., 2020)	71.3	84.1	79.8	92.3	74.0	90.0	24.2	38.5	41.4	51.5	35.9	46.1
+ SapBert	91.0	94.7	93.3	95.5	96.6	97.6	53.0	73.7	72.4	89.1	63.3	77.0
BlueBert (Peng et al., 2019)	75.7	87.2	83.2	91.0	87.7	94.1	41.6	61.9	41.5	48.5	42.9	52.9
+ SapBert	90.9	94.0	93.4	96.0	96.7	98.2	49.6	73.1	72.4	89.4	66.0	78.8
ClinicalBert (Alsentzer et al., 2019)	72.1	84.5	82.7	91.6	75.9	88.5	43.9	54.3	43.1	51.8	40.6	61.8
+ SapBert	91.1	95.1	93.0	95.7	96.6	97.7	51.5	73.0	71.1	88.5	64.3	77.3
SciBert (Beltagy et al., 2019)	85.1	88.4	89.3	92.8	94.2	95.5	42.3	51.9	48.0	54.8	45.8	66.8
+ SapBert	91.7	95.2	93.3	95.7	96.6	98.0	50.1	73.9	72.1	88.7	64.5	77.5
UmlsBert (Michalopoulos et al., 2020)	77.0	85.4	85.5	92.5	88.9	94.1	36.1	55.8	44.4	54.5	44.6	53.0
+ SapBert	91.2	95.2	92.8	95.5	96.6	97.7	52.1	73.2	72.6	89.3	63.4	76.9
PubMedBert (Gu et al., 2020)	77.8	86.9	89.0	93.8	93.0	94.6	43.9	64.7	42.5	49.6	46.8	53.2
+ SapBert	92.0	95.6	93.5	96.0	96.5	98.2	50.8	74.4	70.5	88.9	65.9	77.9
supervised SOTA	91.1	93.9	93.2	96.0	96.6	97.2	OOM	OOM	87.5	-	79.0	-
PubMedBert	77.8	86.9	89.0	93.8	93.0	94.6	43.9	64.7	42.5	49.6	46.8	53.2
+ SapBert	92.0	95.6	93.5	96.0	96.5	98.2	50.8	74.4	70.5	88.9	65.9	77.9
+ SapBert (Adapter$_{13\%}$)	91.5	95.8	93.6	96.3	96.5	98.0	50.7	75.0†	67.5	87.1	64.5	74.9
+ SapBert (Adapter$_{1\%}$)	90.9	95.4	93.8†	96.5†	96.5	97.9	52.2†	74.8	65.7	84.0	63.5	74.2
+ SapBert (Fine-tuned)	92.3	95.5	93.2	95.4	96.5	97.9	50.4	73.9	89.0†	96.2†	75.1 (81.1†)	85.5 (86.1†)
BioSyn	91.1	93.9	93.2	96.0	96.6	97.2	OOM	OOM	82.6	87.0	71.3	77.8
+ (init. w/) SapBert	92.5†	96.2†	93.6	96.2	96.8	98.4†	OOM	OOM	87.6	95.6	77.0	84.2

Table 1: **Top**: Comparison of 7 BERT-based models before and after SapBert pretraining (+ SapBert). All results in this section are from unsupervised learning (not fine-tuned on task data). The gradient of green indicates the improvement comparing to the base model (the deeper the more). **Bottom**: SapBert vs. SOTA results. Blue and red denote unsupervised and supervised models. **Bold** and underline denote the best and second best results in the column. "†" denotes statistically significant better than supervised SOTA (T-test, $\rho < 0.05$). On COMETA, the results inside the parentheses added the supervised SOTA's dictionary back-off technique (Basaldella et al., 2020). "-": not reported in the SOTA paper. "OOM": out-of-memory (192GB+).

figurations specified in App. §B.4). For other hyperparameters used, please view App. §C.2.

Evaluation Data and Protocol. We experiment on 6 different English MEL datasets: 4 in the scientific domain (NCBI, Doğan et al. 2014; BC5CDR-c and BC5CDR-d, Li et al. 2016; MedMentions, Mohan and Li 2018) and 2 in the social media domain (COMETA, Basaldella et al. 2020 and AskAPatient, Limsopatham and Collier 2016). Descriptions of the datasets and their statistics are provided in App. §A. We report Acc$_{@1}$ and Acc$_{@5}$ (denoted as @1 and @5) for evaluating performance. In all experiments, SapBert denotes further pretraining with our self-alignment method on UMLS. At the test phase, for all SapBert models we use nearest neighbour search without further fine-tuning on task data (unless stated otherwise). Except for numbers reported in previous papers, all results are the average of five runs with different random seeds.

Fine-Tuning on Task Data. The red rows in Tab. 1 are results of models (further) fine-tuned on the training sets of the six MEL datasets. Similar to pretraining, a positive pair list is generated through traversing the combinations of mention and all ground truth synonyms where mentions are from

the training set and ground truth synonyms are from the reference ontology. We use the same optimiser and learning rates but train with a batch size of 256 (to accommodate the memory of 1 GPU). On scientific language datasets, we train for 3 epochs while on AskAPatient and COMETA we train for 15 and 10 epochs respectively. For BioSyn on social media language datasets, we empirically found that 10 epochs work the best. Other configurations are the same as the original BioSyn paper.

3.2 Main Results and Analysis

*Bert + SapBert (Tab. 1, top). We illustrate the impact of SapBert pretraining over 7 existing BERT-based models (*Bert = {BioBert, PubMedBert, ...}). SapBert obtains consistent improvement over all *Bert models across all datasets, with larger gains (by up to 31.0% absolute Acc$_{@1}$ increase) observed in the social media domain. While SciBert is the leading model before applying SapBert, PubMedBert+SapBert performs the best afterwards.

SapBert vs. SOTA (Tab. 1, bottom). We take PubMedBert+SapBert (w/wo fine-tuning) and compare against various published SOTA results (see App. §C.1 for a full listing of 10 baselines)

which all require task supervision. For the scientific language domain, the SOTA is BIOSYN (Sung et al., 2020). For the social media domain, the SOTA are Basaldella et al. (2020) and GEN-RANK (Xu et al., 2020) on COMETA and AskAPatient respectively. All these SOTA methods combine BERT with heuristic modules such as tf-idf, string matching and information retrieval system (i.e. Apache Lucene) in a multi-stage manner.

Measured by $Acc_{@1}$, SAPBERT achieves new SOTA with statistical significance on 5 of the 6 datasets and for the dataset (BC5CDR-c) where SAPBERT is not significantly better, it performs on par with SOTA (96.5 vs. 96.6). Interestingly, on scientific language datasets, SAPBERT outperforms SOTA without any task supervision (fine-tuning mostly leads to overfitting and performance drops). On social media language datasets, unsupervised SAPBERT lags behind supervised SOTA by large margins, highlighting the well-documented complex nature of social media language (Baldwin et al., 2013; Limsopatham and Collier, 2015, 2016; Basaldella et al., 2020; Tutubalina et al., 2020). However, after fine-tuning on the social media datasets (using the MS loss introduced earlier), SAPBERT outperforms SOTA significantly, indicating that knowledge acquired during the self-aligning pretraining can be adapted to a shifted domain without much effort.

The ADAPTER Variant. As an option for parameter efficient pretraining, we explore a variant of SAPBERT using a recently introduced training module named ADAPTER (Houlsby et al., 2019). While maintaining the same pretraining scheme with the same SAPBERT online mining + MS loss, instead of training from the full model of PUBMEDBERT, we insert new ADAPTER layers between Transformer layers of the fixed PUBMEDBERT, and only train the weights of these ADAPTER layers. In our experiments, we use the enhanced ADAPTER configuration by Pfeiffer et al. (2020). We include two variants where trained parameters are 13.22% and 1.09% of the full SAPBERT variant. The ADAPTER variant of SAPBERT achieves comparable performance to full-model-tuning in scientific datasets but lags behind in social media datasets, Tab. 1. The results indicate that more parameters are needed in pretraining for knowledge transfer to a shifted domain, in our case, the social media datasets.

The Impact of Online Mining (Eq. (1)). As suggested in Tab. 2, switching off the online hard pairs mining procedure causes a large performance drop in @1 and a smaller but still significant drop in @5. This is due to the presence of many easy and already well-separated samples in the mini-batches. These uninformative training examples dominated the gradients and harmed the learning process.

configuration	@1	@5
Mining switched-on	**67.2**	**80.3**
Mining switched-off	$52.3_{\downarrow14.9}$	$76.1_{\downarrow4.2}$

Table 2: This table compares PUBMED-BERT+SAPBERT's performance with and without online hard mining on COMETA (zeroshot general).

Integrating SAPBERT in Existing Systems. SAPBERT can be easily inserted into existing BERT-based MEL systems by initialising the systems with SAPBERT pretrained weights. We use the SOTA scientific language system, BIOSYN (originally initialised with BIOBERT weights), as an example and show the performance is boosted across all datasets (last two rows, Tab. 1).

4 Conclusion

We present SAPBERT, a self-alignment pretraining scheme for learning biomedical entity representations. We highlight the consistent performance boost achieved by SAPBERT, obtaining new SOTA in all six widely used MEL benchmarking datasets. Strikingly, without any fine-tuning on task-specific labelled data, SAPBERT already outperforms the previous supervised SOTA (sophisticated hybrid entity linking systems) on multiple datasets in the scientific language domain. Our work opens new avenues to explore for general domain self-alignment (e.g. by leveraging knowledge graphs such as DBpedia). We plan to incorporate other types of relations (i.e., hypernymy and hyponymy) and extend our model to sentence-level representation learning. In particular, our ongoing work using a combination of SAPBERT and ADAPTER is a promising direction for tackling sentence-level tasks.

Acknowledgements

References

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

Timothy Baldwin, Paul Cook, Marco Lui, Andrew MacKinlay, and Li Wang. 2013. How noisy social media text, how diffrnt social media sources? In *Proceedings of the Sixth International Joint Conference on Natural Language Processing (IJCNLP)*, pages 356–364, Nagoya, Japan. Asian Federation of Natural Language Processing.

Marco Basaldella, Fangyu Liu, Ehsan Shareghi, and Nigel Collier. 2020. COMETA: A corpus for medical entity linking in the social media. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3122–3137, Online. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Olivier Bodenreider. 2004. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32:D267–D270.

Allan Peter Davis, Cynthia J Grondin, Robin J Johnson, Daniela Sciaky, Roy McMorran, Jolene Wiegers, Thomas C Wiegers, and Carolyn J Mattingly. 2019. The comparative toxicogenomics database: update 2019. *Nucleic Acids Research*, 47:D948–D954.

Allan Peter Davis, Thomas C Wiegers, Michael C Rosenstein, and Carolyn J Mattingly. 2012. MEDIC: a practical disease vocabulary used at the comparative toxicogenomics database. *Database*.

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 (NAACL), Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. NCBI disease corpus: a resource for disease name recognition and concept normalization. *Journal of Biomedical Informatics*, 47:1–10.

Kevin Donnelly. 2006. SNOMED-CT: The advanced terminology and coding system for eHealth. *Studies in health technology and informatics*, 121:279.

Jennifer D'Souza and Vincent Ng. 2015. Sieve-based entity linking for the biomedical domain. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP) (Volume 2: Short Papers)*, pages 297–302, Beijing, China. Association for Computational Linguistics.

Jacob Goldberger, Geoffrey E Hinton, Sam T Roweis, and Russ R Salakhutdinov. 2005. Neighbourhood components analysis. In *Advances in Neural Information Processing Systems*, pages 513–520.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2020. Domain-specific language model pretraining for biomedical natural language processing. *arXiv:2007.15779*.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Zongcheng Ji, Qiang Wei, and Hua Xu. 2020. BERT-based ranking for biomedical entity normalization. *AMIA Summits on Translational Science Proceedings*, 2020:269.

Donghyeon Kim, Jinhyuk Lee, Chan Ho So, Hwisang Jeon, Minbyul Jeong, Yonghwa Choi, Wonjin Yoon, Mujeen Sung, , and Jaewoo Kang. 2019. A neural named entity recognition and multi-type normalization tool for biomedical text mining. *IEEE Access*, 7:73729–73740.

Robert Leaman and Zhiyong Lu. 2016. TaggerOne: joint named entity recognition and normalization with semi-markov models. *Bioinformatics*, 32:2839–2846.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: a pretrained biomedical language representation model

for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Sunwon Lee, Donghyeon Kim, Kyubum Lee, Jaehoon Choi, Seongsoon Kim, Minji Jeon, Sangrak Lim, Donghee Choi, Sunkyu Kim, Aik-Choon Tan, et al. 2016. BEST: next-generation biomedical entity search tool for knowledge discovery from biomedical literature. *PloS one*, 11:e0164680.

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

Nut Limsopatham and Nigel Collier. 2015. Adapting phrase-based machine translation to normalise medical terms in social media messages. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1675–1680, Lisbon, Portugal. Association for Computational Linguistics.

Nut Limsopatham and Nigel Collier. 2016. Normalising medical concepts in social media texts by learning semantic representation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1014–1023.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(Nov):2579–2605.

George Michalopoulos, Yuanxin Wang, Hussam Kaka, Helen Chen, and Alex Wong. 2020. Umlsbert: Clinical domain knowledge augmentation of contextual embeddings using the unified medical language system metathesaurus. *arXiv preprint arXiv:2010.10391*.

Sunil Mohan and Donghui Li. 2018. MedMentions: A large biomedical corpus annotated with UMLS concepts. In *Automated Knowledge Base Construction*.

Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. 2016. Deep metric learning via lifted structured feature embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4004–4012.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of bert and elmo on ten benchmarking datasets. In *Proceedings of the 2019 Workshop on Biomedical Natural Language Processing*, pages 58–65.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.

Minh C Phan, Aixin Sun, and Yi Tay. 2019. Robust representation learning of biomedical names. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3275–3285.

Kirk Roberts, Matthew S Simpson, Ellen M Voorhees, and William R Hersh. 2015. Overview of the trec 2015 clinical decision support track. In *TREC*.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823.

Elliot Schumacher, Andriy Mulyar, and Mark Dredze. 2020. Clinical concept linking with contextualized neural representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8585–8592.

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

Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. 2020. Circle loss: A unified perspective of pair similarity optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6398–6407.

Mujeen Sung, Hwisang Jeon, Jinhyuk Lee, and Jaewoo Kang. 2020. Biomedical entity representations with synonym marginalization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 3641–3650, Online. Association for Computational Linguistics.

Elena Tutubalina, Artur Kadurin, and Zulfat Miftahutdinov. 2020. Fair evaluation in concept normalization: a large-scale comparative analysis for bert-based models. In *Proceedings of the 28th International Conference on Computational Linguistics (COLING)*.

Elena Tutubalina, Zulfat Miftahutdinov, Sergey Nikolenko, and Valentin Malykh. 2018. Medical concept normalization in social media posts with recurrent neural networks. *Journal of Biomedical Informatics*, 84:93–102.

Ivan Vulić, Edoardo Maria Ponti, Robert Litschko, Goran Glavaš, and Anna Korhonen. 2020. Probing pretrained language models for lexical semantics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7222–7240, Online. Association for Computational Linguistics.

Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. 2019. Multi-similarity loss with general pair weighting for deep metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5022–5030.

Yanshan Wang, Sijia Liu, Naveed Afzal, Majid Rastegar-Mojarad, Liwei Wang, Feichen Shen, Paul Kingsbury, and Hongfang Liu. 2018. A comparison of word embeddings for the biomedical natural language processing. *Journal of Biomedical Informatics*, 87:12–20.

Dustin Wright, Yannis Katsis, Raghav Mehta, and Chun-Nan Hsu. 2019. Normco: Deep disease normalization for biomedical knowledge base construction. In *Automated Knowledge Base Construction*.

Dongfang Xu, Zeyu Zhang, and Steven Bethard. 2020. A generate-and-rank framework with semantic type regularization for biomedical concept normalization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8452–8464.

A Evaluation Datasets Details

We divide our experimental datasets into two categories (1) scientific language datasests where the data is extracted from scientific papers and (2) social media language datasets where the data is coming from social media forums like Reddit.com. For an overview of the key statistics, see Tab. 3.

A.1 Scientific Language Datasets

NCBI disease (Doğan et al., 2014) is a corpus containing 793 fully annotated PubMed abstracts and 6,881 mentions. The mentions are mapped into the MEDIC dictionary (Davis et al., 2012). We denote this dataset as "NCBI" in our experiments.

BC5CDR (Li et al., 2016) consists of 1,500 PubMed articles with 4,409 annotated chemicals, 5,818 diseases and 3,116 chemical-disease interactions. The disease mentions are mapped into the MEDIC dictionary like the NCBI disease corpus.

The chemical mentions are mapped into the Comparative Toxicogenomics Database (CTD) (Davis et al., 2019) chemical dictionary. We denote the disease and chemical mention sets as "BC5CDR-d" and "BC5CDR-c" respectively. For NCBI and BC5CDR we use the same data and evaluation protocol by Sung et al. (2020).[11]

MedMentions (Mohan and Li, 2018) is a very-large-scale entity linking dataset containing over 4,000 abstracts and over 350,000 mentions linked to UMLS 2017AA. According to Mohan and Li (2018), training TAGGERONE (Leaman and Lu, 2016), a very popular MEL system, on a subset of MedMentions require >900 GB of RAM. Its massive number of mentions and more importantly the used reference ontology (UMLS 2017AA has 3M+ concepts) make the application of most MEL systems infeasible. However, through our metric learning formulation, SAPBERT can be applied on MedMentions with minimal effort.

A.2 Social-Media Language Datasets

AskAPatient (Limsopatham and Collier, 2016) includes 17,324 adverse drug reaction (ADR) annotations collected from askapatient.com blog posts. The mentions are mapped to 1,036 medical concepts grounded onto SNOMED-CT (Donnelly, 2006) and AMT (the Australian Medicines Terminology). For this dataset, we follow the 10-fold evaluation protocol stated in the original paper.[12]

COMETA (Basaldella et al., 2020) is a recently released large-scale MEL dataset that specifically focuses on MEL in the social media domain, containing around 20k medical mentions extracted from health-related discussions on reddit.com. Mentions are mapped to SNOMED-CT. We use the "stratified (general)" split and follow the evaluation protocol of the original paper.[13]

B Model & Training Details

B.1 The Choice of Base Models

We list all the versions of BERT models used in this study, linking to the specific versions in Tab. 5. Note that we exhaustively tried all official variants of the selected models and the best performing ones are chosen. All BERT models refer to the BERT$_{Base}$ architecture in this paper.

---

[11]https://github.com/dmis-lab/BioSyn
[12]https://zenodo.org/record/55013
[13]https://www.siphs.org/corpus

dataset	NCBI	BC5CDR-d	BC5CDR-c	MedMentions	AskAPAtient	COMETA (s.g.)	COMETA (z.g.)
Ontology	MEDIC	MEDIC	CTD	UMLS 2017AA	SNOMED & AMT	SNOMED	SNOMED
$\mathcal{C}_{searched} \subsetneq \mathcal{C}_{ontology}$?	✗	✗	✗	✗	✓	✗	✗
$|\mathcal{C}_{searched}|$	11,915	11,915	171,203	3,415,665	1,036	350,830	350,830
$|\mathcal{S}_{searched}|$	71,923	71,923	407,247	14,815,318	1,036	910,823	910,823
$|\mathcal{M}_{train}|$	5,134	4,182	5,203	282,091	15,665.2	13,489	14,062
$|\mathcal{M}_{validation}|$	787	4,244	5,347	71,062	792.6	2,176	1,958
$|\mathcal{M}_{test}|$	960	4,424	5,385	70,405	866.2	4,350	3,995

Table 3: This table contains basic statistics of the MEL datasets used in the study. $\mathcal{C}$ denotes the set of concepts; $\mathcal{S}$ denotes the set of all surface forms / synonyms of all concepts in $\mathcal{C}$; $\mathcal{M}$ denotes the set of mentions / queries. COMETA (s.g.) and (z.g.) are the stratified (general) and zeroshot (general) split respectively.

model	NCBI		BC5CDR-d		BC5CDR-c		MedMentions		AskAPatient		COMETA	
	@1	@5	@1	@5	@1	@5	@1	@5	@1	@5	@1	@5
SIEVE-BASED (D'Souza and Ng, 2015)	84.7	-	84.1	-	90.7	-	-	-	-	-	-	-
WORDCNN (Limsopatham and Collier, 2016)	-	-	-	-	-	-	-	-	81.4	-	-	-
WORDGRU+TF-IDF (Tutubalina et al., 2018)	-	-	-	-	-	-	-	-	85.7	-	-	-
TAGGERONE (Leaman and Lu, 2016)	87.7	-	88.9	-	94.1	-	OOM	OOM	-	-	-	-
NORMCO (Wright et al., 2019)	87.8	-	88.0	-	-	-	-	-	-	-	-	-
BNE (Phan et al., 2019)	87.7	-	90.6	-	95.8	-	-	-	-	-	-	-
BERTRANK (Ji et al., 2020)	89.1	-	-	-	-	-	-	-	-	-	-	-
GEN-RANK (Xu et al., 2020)	-	-	-	-	-	-	-	-	**87.5**	-	-	-
BIOSYN (Sung et al., 2020)	**91.1**	**93.9**	**93.2**	**96.0**	**96.6**	**97.2**	OOM	OOM	82.6*	87.0*	71.3*	77.8*
DICT+SOILOS+NEURAL (Basaldella et al., 2020)	-	-	-	-	-	-	-	-	-	-	**79.0**	-
supervised SOTA	91.1	93.9	93.2	96.0	96.6	97.2	OOM	OOM	87.5	-	79.0	-

Table 4: A list of baselines on the 6 different MEL datasets, including both scientific and social media language ones. The last row collects reported numbers from the best performing models. "∗" denotes results produced using official released code. "-" denotes results not reported in the cited paper. "OOM" means out-of-memoery.

B.2 Comparing Loss Functions

We use COMETA (zeroshot general) as a benchmark for selecting learning objectives. Note that this split of COMETA is different from the stratified-general split used in Tab. 4. It is very challenging (so easy to see the difference of the performance) and also does not directly affect the model's performance on other datasets. The results are listed in Tab. 6. Note that online mining is switched on for all models here.

loss	@1	@5
cosine loss (Phan et al., 2019)	55.1	64.6
max-margin triplet loss (Basaldella et al., 2020)	64.6	74.6
NCA loss (Goldberger et al., 2005)	65.2	77.0
Lifted-Structure loss (Oh Song et al., 2016)	62.0	72.1
InfoNCE (Oord et al., 2018; He et al., 2020)	63.3	74.2
Circle loss (Sun et al., 2020)	66.7	78.7
Multi-Similarity loss (Wang et al., 2019)	**67.2**	**80.3**

Table 6: This table compares loss functions used for SAPBERT pretraining. Numbers reported are on COMETA (zeroshot general).

The cosine loss was used by Phan et al. (2019) for learning UMLS synonyms for LSTM models. The max-margin triplet loss was used by Basaldella et al. (2020) for training MEL models. A very similar (though not identical) hinge-loss was used by Schumacher et al. (2020) for clinical concept linking. InfoNCE has been very popular in self-supervised learning and contrastive learning (Oord et al., 2018; He et al., 2020). Lifted-Structure loss (Oh Song et al., 2016) and NCA loss (Goldberger et al., 2005) are two very classic metric learning objectives. Multi-Similarity loss (Wang et al., 2019) and Circle loss (Sun et al., 2020) are two recently proposed metric learning objectives and have been considered as SOTA on large-scale visual recognition benchmarks.

B.3 Details of ADAPTERs

In Tab. 7 we list number of parameters trained in the three ADAPTER variants along with full-model-tuning for easy comparison.

model	URL
vanilla BERT (Devlin et al., 2019)	https://huggingface.co/bert-base-uncased
BIOBERT (Lee et al., 2020)	https://huggingface.co/dmis-lab/biobert-v1.1
BLUEBERT (Peng et al., 2019)	https://huggingface.co/bionlp/bluebert_pubmed_mimic_uncased_L-12_H-768_A-12
CLINICALBERT (Alsentzer et al., 2019)	https://huggingface.co/emilyalsentzer/Bio_ClinicalBERT
SCIBERT (Beltagy et al., 2019)	https://huggingface.co/allenai/scibert_scivocab_uncased
UMLSBERT (Michalopoulos et al., 2020)	https://www.dropbox.com/s/qaoq5gfen69xdcc/umlsbert.tar.xz?dl=0
PUBMEDBERT (Gu et al., 2020)	https://huggingface.co/microsoft/BiomedNLP-PubMedBERT-base-uncased-abstract-fulltext

Table 5: This table lists the URL of models used in this study.

method	reduction rate	#params	$\frac{\text{\#params}}{\text{\#params in BERT}}$
ADAPTER$_{13\%}$	1	14.47M	13.22%
ADAPTER$_{1\%}$	16	0.60M	1.09%
full-model-tuning	-	109.48M	100%

Table 7: This table compares number of parameters trained in ADAPTER variants and also full-model-tuning.

B.4 Hardware Configurations

All our experiments are conducted on a server with specifications listed in Tab. 8.

hardware	specification
RAM	192 GB
CPU	Intel Xeon W-2255 @3.70GHz, 10-core 20-threads
GPU	NVIDIA GeForce RTX 2080 Ti (11 GB) $\times$ 4

Table 8: Hardware specifications of the used machine.

C Other Details

C.1 The Full Table of Supervised Baseline Models

The full table of supervised baseline models is provided in Tab. 4.

C.2 Hyper-Parameters Search Scope

Tab. 9 lists hyper-parameter search space for obtaining the set of used numbers. Note that the chosen hyper-parameters yield the overall best performance but might be sub-optimal on any single dataset. Also, we balanced the memory limit and model performance.

C.3 A High-Resolution Version of Fig. 1

We show a clearer version of t-SNE embedding visualisation in Fig. 3.

hyper-parameters	search space
learning rate for pretraining & fine-tuning SAPBERT	{1e-4, 2e-5*, 5e-5, 1e-5, 1e-6}
pretraining batch size	{128, 256, 512*, 1024}
pretraining training iterations	{10k, 20k, 30k, 40k, 50k (1 epoch)*, 100k (2 epochs)}
fine-tuning epochs on scientific language datasets	{1, 2, 3*, 5}
fine-training epochs on AskAPatient	{5, 10, 15*, 20}
fine-training epochs on COMETA	{5, 10*, 15, 20}
max_seq_length of BERT tokenizer	{15, 20, 25*, 30}
λ in Online Mining	{-0.05, -0.1, -0.2*, -0.3}
α in MS loss	{1, 2 (Wang et al., 2019)*, 3}
β in MS loss	{40, 50 (Wang et al., 2019)*, 60}
ϵ in MS loss	{0.5*, 1 (Wang et al., 2019)}
α in max-margin triplet loss	{0.05, 0.1, 0.2 (Basaldella et al., 2020)*, 0.3}
softmax scale in NCA loss	{1 (Goldberger et al., 2005), 5, 10, 20*, 30}
α in Lifted-Structured loss	{0.5*, 1 (Oh Song et al., 2016)}
τ (temperature) in InfoNCE	{0.07 (He et al., 2020)*, 0.5 (Oord et al., 2018)}
m in Circle loss	{0.25 (Sun et al., 2020)*, 0.4 (Sun et al., 2020)}
γ in Circle loss	{80 (Sun et al., 2020), 256 (Sun et al., 2020)*}

Table 9: This table lists the search space for hyper-parameters used. ∗ means the used ones for reporting results.
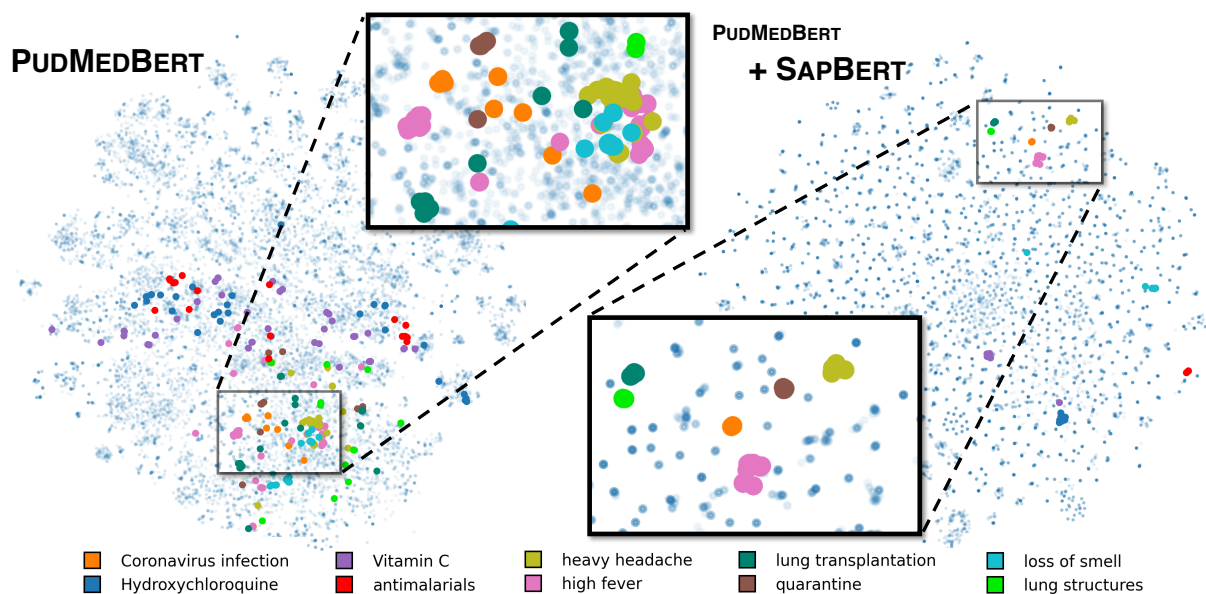

Figure 3: Same as Fig. 1 in the main text, but generated with a higher resolution.