Triplet-Trained Vector Space and Sieve-Based Search Improve Biomedical Concept Normalization

Dongfang Xu and **Steven Bethard**
School of Information
University of Arizona
Tucson, AZ
{dongfangxu9,bethard}@email.arizona.edu

Abstract

Concept normalization, the task of linking textual mentions of concepts to concepts in an ontology, is critical for mining and analyzing biomedical texts. We propose a vector-space model for concept normalization, where mentions and concepts are encoded via transformer networks that are trained via a triplet objective with online hard triplet mining. The transformer networks refine existing pre-trained models, and the online triplet mining makes training efficient even with hundreds of thousands of concepts by sampling training triples within each mini-batch. We introduce a variety of strategies for searching with the trained vector-space model, including approaches that incorporate domain-specific synonyms at search time with no model retraining. Across five datasets, our models that are trained only once on their corresponding ontologies are within 3 points of state-of-the-art models that are retrained for each new domain. Our models can also be trained for each domain, achieving new state-of-the-art on multiple datasets.

1 Introduction

Concept normalization (aka. entity linking or entity normalization) is a fundamental task of information extraction which aims to map concept mentions in text to standard concepts in a knowledge base or ontology. This task is important for mining and analyzing unstructured text in the biomedical domain as the texts describing biomedical concepts have many morphological and orthographical variations, and utilize different word orderings or equivalent words. For instance, *heart attack*, *coronary attack*, *MI*, *myocardial infarction*, *cardiac infarction*, and *cardiovascular stroke* all refer to the same concept. Linking such terms with their corresponding concepts in an ontology or knowledge base is critical for data interoperability and the development of natural language processing (NLP) techniques.

Research on concept normalization has grown thanks to shared tasks such as disorder normalization in the 2013 ShARe/CLEF (Suominen et al., 2013), chemical and disease normalization in BioCreative V Chemical Disease Relation (CDR) Task (Wei et al., 2015), and medical concept normalization in 2019 n2c2 shared task (Henry et al., 2020), and to the availability of annotated data (Doğan et al., 2014; Luo et al., 2019). Existing approaches can be divided into three categories: rule-based approaches using string-matching or dictionary look-up (Leal et al., 2015; D'Souza and Ng, 2015; Lee et al., 2016), which rely heavily on handcrafted rules and domain knowledge; supervised multi-class classifiers (Limsopatham and Collier, 2016; Lee et al., 2017; Tutubalina et al., 2018; Niu et al., 2019; Li et al., 2019), which cannot generalize to concept types not present in their training data; and two-step frameworks based on a non-trained candidate generator and a supervised candidate ranker (Leaman et al., 2013; Li et al., 2017; Liu and Xu, 2017; Nguyen et al., 2018; Murty et al., 2018; Mondal et al., 2019; Ji et al., 2020; Xu et al., 2020), which require complex pipelines and fail if the candidate generator does not find the gold truth concept.

We propose a vector space model for concept normalization, where mentions and concepts are encoded as vectors – via transformer networks trained via a triplet objective with online hard triplet mining – and mentions are matched to concepts by vector similarity. The online hard triplet mining strategy selects the hard positive/negative exemplars from within a mini-batch during training, which ensures consistently increasing difficulty of triplets as the network trains for fast convergence. There are two advantages of applying the vector space model for concept normalization: 1) it is computationally cheap compared with other supervised classification approaches as we only compute the representations for all concepts in ontology once

Proceedings of the BioNLP 2021 workshop, pages 11–22
June 11, 2021.

after training the network; 2) it allows concepts and synonyms to be added or deleted after the network is trained, a flexibility that is important for the biomedical domain where frequent updates to ontologies like the Unified Medical Language System (UMLS) Metathesaurus[1] are common. Unlike prior work, our simple and efficient model requires neither negative sampling before the training nor a candidate generator during inference.

Our work makes the following contributions:

- We propose a triplet network with online hard triplet mining for training a vector-space model for concept normalization, a simpler and more efficient approach than prior work.
- We propose and explore a variety of strategies for matching mentions to concepts using the vector-space model, with the most successful being a simple sieve-based approach that checks domain-specific synonyms before domain-independent ones.
- Our framework produces models trained on only the ontology – no domain-specific training – that can incorporate domain-specific concept synonyms at search time without retraining, and these models achieve within 3 points of state-of-the-art on five datasets.
- Our framework also allows models to be trained for each domain, achieving state-of-the-art performance on multiple datasets.

The code for our proposed framework is available at `https://github.com/dongfang91/Triplet-Search-ConNorm`.

2 Related work

Earlier work on concept normalization focuses on how to use morphological information to conduct lexical look-up and string matching (Kang et al., 2013; D'Souza and Ng, 2015; Leaman et al., 2015; Leal et al., 2015; Kate, 2016; Lee et al., 2016; Jonnagaddala et al., 2016). They rely heavily on handcrafted rules and domain knowledge, e.g., D'Souza and Ng (2015) define 10 types of rules at different priority levels to measure morphological similarity between mentions and candidate concepts in the ontologies. The lack of lexical overlap between concept mention and concept in domains like social media, makes rule-based approaches that rely on lexical matching less applicable.

Supervised approaches for concept normalization have improved with the availability of annotated data and deep learning techniques. When the number of concepts to be predicted is small, classification-based approaches (Limsopatham and Collier, 2016; Lee et al., 2017; Tutubalina et al., 2018; Niu et al., 2019; Li et al., 2019; Miftahutdinov and Tutubalina, 2019) are often adopted, with the size of the classifier's output space equal to the number of concepts. Approaches differ in neural architectures, such as character-level convolution neural networks (CNN) with multi-task learning (Niu et al., 2019) and pre-trained transformer networks (Li et al., 2019; Miftahutdinov and Tutubalina, 2019). However, classification approaches struggle when the annotated training data does not contain examples of all concepts – common when there are many concepts in the ontology – since the output space of the classifier will not include concepts absent from the training data.

To alleviate the problems of classification-based approaches, researchers apply learning to rank in concept normalization, a two-step framework including a non-trained candidate generator and a supervised candidate ranker that takes both mention and candidate concept as input. Previous candidate rankers have used point-wise learning to rank (Li et al., 2017), pair-wise learning to rank (Leaman et al., 2013; Liu and Xu, 2017; Nguyen et al., 2018; Mondal et al., 2019), and list-wise learning to rank (Murty et al., 2018; Ji et al., 2020; Xu et al., 2020). These learning to rank approaches also have drawbacks. Firstly, if the candidate generator fails to produce the gold truth concept, the candidate ranker will also fail. Secondly, the training of candidate ranker requires negative sampling beforehand, and it is unclear if these pre-selected negative samples are informative for the whole training process (Hermans et al., 2017; Sung et al., 2020).

Inspired by Schroff et al. (2015), we propose a triplet network with online hard triplet mining for concept normalization. Our framework sets up concept normalization as a one-step process, calculating similarity between vector representations of the mention and of all concepts in the ontology. Online hard triplet mining allows such a vector space model to generate triplets of (mention, true concept, false concept) within a mini-batch, leading to efficient training and fast convergence (Schroff et al., 2015). In contrast with previous vector space models where mention and candidate

[1] `https://www.nlm.nih.gov/research/umls/knowledge_sources/metathesaurus/index.html`

concepts are mapped to vectors via TF-IDF (Leaman et al., 2013), TreeLSTMs (Liu and Xu, 2017), CNNs (Nguyen et al., 2018; Mondal et al., 2019) or ELMO (Schumacher et al., 2020), we generate vector representations with BERT (Devlin et al., 2019), since it can encode both surface and semantic information (Ma et al., 2019).

There are a few similar works to our vector space model, CNN-triplet (Mondal et al., 2019), BIOSYN (Sung et al., 2020), RoBERTa-Node2Vec (Pattisapu et al., 2020), and TTI (Henry et al., 2020). CNN-triplet is a two-step approach, requiring a generator to generate candidates for training the triplet network, and requiring various embedding resources as input to CNN-based encoder. BIOSYN, RoBERTa-Node2Vec, and TTI are one-step approaches. BIOSYN requires an iterative candidate retrieval over the entire training data during each training step, requires both BERT-based and TF-IDF-based representations, and performs a variety of pre-processing such as acronym expansion. Both RoBERTa-Node2Vec and TTI use a BERT-based encoder to encode the mention texts into a vector space, but they differ in how to generate vector representations for medical concepts. Specifically, RoBERTa-Node2Vec uses a Node2Vec graph embedding approach to generate concept representations, and fixes such representations during training, while TTI randomly initializes vector representations for concepts, and keeps such representations learnable during training. Note that none of these works explore search strategies that allow domain-specific synonyms to be added without retraining the model, while we do.

3 Proposed methods

We define a concept mention m as a text string in a corpus D, and a concept c as a unique identifier in an ontology O. The goal of concept normalization is to find a mapping function f that maps each textual mention to its correct concept, i.e., $c = f(m)$. We define concept text t as a text string denoting the concept c, and $t \in T(c)$, where $T(c)$ is all the concept texts denoting concept c. Concept text may come from an ontology, $t \in O(c)$, where $O(c)$ is the synonyms of the concept c from the ontology O, or from an annotated corpus, $t \in D(c)$, where $D(c)$ is the mentions of the concept c in an annotated corpus D. $T(c)$ will allow the generation of tuples (t, c) such as (*MI,C0027051*) and (*Myocardial Infarction,C0027051*). Note that, for a

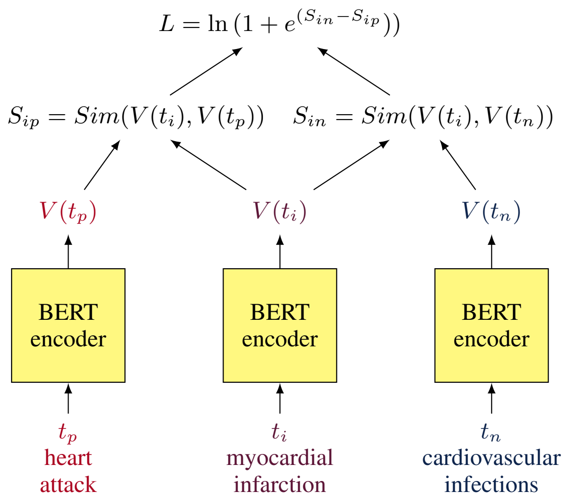

Figure 1: Example of loss calculation for a single instance of triplet-based training. The same BERT model is used for encoding t_i, t_p, and t_n.

concept c, it is common to have $|O(c)| > |D(c)|$, $O(c) \cap D(c) = \emptyset$, or even $D(c) = \emptyset$, i.e., it is common for there to be more concept synonyms in the ontology than the annotated corpus, it is common for the ontology and annotated corpus to provide different concept synonyms, and it is common that annotated corpus only covers a small subset of all concepts in an ontology.

We implement f as a vector space model:

$$f(m) = \operatorname*{argmax}_{\substack{c \in O \\ t \in T(c)}} Sim(V(m), V(t)) \quad (1)$$

where $V(x)$ is a vector representation of text x and Sim is a similarity measure such as cosine similarity, inner product, or euclidean distance. We learn the vector representations $V(x)$ using a triplet network architecture (Hoffer and Ailon, 2015), which learns from triplets of (anchor text t_i, positive text t_p, negative text t_n) where t_i and t_p are texts for the same concept, and t_n is a text for a different concept. The triplet network attempts to learn V such that for all training triplets:

$$Sim(V(t_i), V(t_{ip})) > Sim(V(t_i), V(t_{in})) \quad (2)$$

The triplet network architecture has been adopted in learning representations for images (Schroff et al., 2015; Gordo et al., 2016) and text (Neculoiu et al., 2016; Reimers and Gurevych, 2019). It consists of three instances of the same sub-network (with shared parameters). When fed a (t_i, t_{ip}, t_{in}) triplet of texts, the sub-network outputs vector representations for each text, which are then fed into a triplet loss. We adopt PubMed-BERT (Gu et al.,

2020) as the sub-network, where the representation for the concept text is an average pooling of the representations for all sub-word tokens[2]. This architecture is shown in Figure 1. The inputs to our model are only the mentions or synonyms. We leave the resolution of ambiguous mentions, which will require exploration of contextual information, for future work.

3.1 Online hard triplet mining

An essential part of learning using triplet loss is how to generate triplets. As the number of synonyms gets larger, the number of possible triplets grows cubically, making training impractical. We follow the idea of online triplet mining (Schroff et al., 2015) which considers only triplets within a mini-batch. We first feed a mini-batch of b concept texts to the PubMed-BERT encoder to generate a d-dimensional representation for each concept text, resulting in a matrix $M \in \mathbb{R}^{b \times d}$. We then compute the pairwise similarity matrix:

$$S = Sim(M, M^T) \tag{3}$$

where each entry S_{ij} corresponds to the similarity score between the i^{th} and j^{th} concept texts in the mini-batch. As the easy triplets would not contribute to the training and result in slower convergence (Schroff et al., 2015), for each concept text t_i, we only select a hard positive t_p and a hard negative t_n from the mini-batch such that:

$$p = \operatorname*{argmin}_{j \in [1,b]: j \neq i \wedge C(j) = C(i)} S_{ij} \tag{4}$$

$$n = \operatorname*{argmax}_{k \in [1,b]: k \neq i \wedge C(k) \neq C(i)} S_{ik} \tag{5}$$

where $C(x)$ is the ontology concept from which t_x was taken, i.e., if $t_x \in T(c)$ then $C(x) = c$.

We train the triplet network using batch hard soft margin loss (Hermans et al., 2017):

$$L(i) = \ln\left(1 + e^{(S_{in} - S_{ip})}\right) \tag{6}$$

where S, n, and p are as in eqs. (3) to (5), and the hinge function, $\max(\cdot, 0)$, in the traditional triplet loss is replaced by a softplus function, $\ln(1 + e^{(\cdot)})$.

3.2 Similarity search

Once our vector space model has been trained, we consider several options for how to find the most similar concept c to a text mention m. First, we must choose a search target: we can search over the concepts from the ontology, or the training data, or both. Second we must choose a representation type: we can compare m directly to each text (ontology synonym or training data mention) of each concept, or we can calculate a vector representation of each concept and then compare m directly to the concept vector. Table 1 summarizes these options.

	Searching Over		Representation Type	
	Ontology	Training Data	Text	Concept
O-T	✓		✓	
O-C	✓			✓
D-T		✓	✓	
D-C		✓		✓
OD-T	✓	✓	✓	
OD-C	✓	✓		✓

Table 1: Names for similarity search modules.

We consider the following search targets:

Data We search over the concepts in the annotated data. These mentions will be more domain-specific (e.g., *PT* may refer to *patient* in clinical notes, but to *physical therapy* in scientific articles), but may be more predictive if the evaluation data is from the same domains. We search over the train subset of the data for dev evaluation, and train + dev subset for test evaluation.

Ontology We search over the concepts in the ontology. The synonyms will be more domain-independent, and the ontology will cover concepts never seen in the annotated training data.

Data and ontology We search over the concepts in both the training data and the ontology. For concepts in the annotated training data, their representations are averaged over mentions in the training data and synonyms in the ontology.

We consider the following representation types:

Text We represent each text (ontology synonym or training data mention) as a vector by running it through our triplet-fine-tuned PubMed-BERT encoder. Concept normalization then compares the mention vector to each text vector:

$$f(m) = \operatorname*{argmax}_{\substack{c \in O \\ t \in T(c)}} Sim(V(m), V(t)) \tag{7}$$

When a retrieved text t is present in more than one concept (e.g., *no appetite* appears in concepts *C0426579*, *C0003123*, *C1971624*), and thus we see the same Sim for multiple concepts, we pick a concept randomly to break ties.

[2] We also experimented with using the output of the *CLS*-token, and max-pooling of the output representations for the sub-word tokens as proposed by (Reimers and Gurevych, 2019), but neither resulted in better performance.

First component	Second component
D-T	O-T
D-T	O-C
D-C	O-T
D-C	O-C
D-T	OD-T
D-T	OD-C
D-C	OD-T
D-C	OD-C

Table 2: Options for components in sieve-based search.

Concept We represent each concept as a vector by taking an average over the triplet-fine-tuned PubMed-BERT representations of that concept's texts (ontology synonyms and/or training data mentions). Concept normalization then compares the mention vector to each concept vector:

$$f(m) = \operatorname*{argmax}_{c \in O} Sim\left(V(m), \operatorname*{mean}_{t \in T(c)} V(t)\right) \quad (8)$$

The averages here mean that different concepts with some (but not all) overlapping synonyms (e.g., *C0426579*, *C0003123*, *C1971624* in UMLS all have the synonym *no appetite*) will end up with different vector representations.

3.2.1 Sieve-based search

Traditional sieve-based approaches for concept normalization (D'Souza and Ng, 2015; Jonnagaddala et al., 2016; Luo et al., 2019; Henry et al., 2020) achieved competitive performance by ordering a sequence of searches over dictionaries from most precise to least precise.

Inspired by this work, we consider a sieve-based similarity search that: 1) searches over the annotated training data, then 2) searches over the ontology (possibly combined with the annotated training data). Table 2 lists all possible combinations of first and second components in sieve-based search. For instance, in sieve-based search **D-T + O-C**, we first search over the annotated corpus using training-data-mention vectors (D-T), and then search over the ontology using concept vectors (O-C).

4 Experiments

4.1 Datasets

We conduct experiments on three scientific article datasets – NCBI (Doğan et al., 2014), BC5CDR-D and BC5CDR-C (Li et al., 2016) – and two clinical note datasets – MCN (Luo et al., 2019) and ShARe/CLEF (Suominen et al., 2013). The statistics of each dataset are described in table 3.

NCBI The NCBI disease corpus[3] contains 17,324 manually annotated disorder mentions from 792 PubMed abstracts. The disorder mentions are mapped to 750 MEDIC lexicon (Davis et al., 2012) concepts. We split the released training set into use 5,134 training mentions and 787 development mentions, and keep the 960 mentions from the original test set as evaluation. We use the 2012 version of MEDIC ontology which contains 11,915 concepts and 71,923 synonyms.

BC5CDR-D & BC5CDR-C These corpora were used in the BioCreative V chemical-induced disease (CID) relation extraction challenge[4]. BC5CDR-D and BC5CDR-C contain 12,850 disease mentions and 15,935 chemical mentions, respectively. The annotated disease mentions are mapped to 1075 unique concepts out of 11,915 concepts in the 2012 version of MEDIC ontology. The chemical mentions are mapped to 1164 unique concepts out of 171,203 concepts from the 2019 version of Comparative Toxicogenomics Database (CTD) chemical ontology. We use the configuration in the BioCreative V challenge to keep the same train/dev/test splits.

ShARe/CLEF The ShARe/CLEF corpus is from the ShARe/CLEF eHealth 2013 Challenge[5], where 11,167 disorder mentions in 298 clinical notes are annotated with their concepts mapping to the 12,6524 disorder concepts from the SNOMED-CT subset of the 2011AA version of UMLS. We take the 199 clinical notes consisting of 5,816 mentions as the train set and 5,351 mentions from the 99 clinical notes as test. Around 30.4% of the mentions in the corpus could not be mapped to any concepts in the ontology, and are assigned the *CUI-less* label.

MCN The MCN corpus from 2019 n2c2 Shared-Task track 3[6] consists of 13,609 concept mentions in 100 discharge summaries. The mentions are mapped to 3,792 unique concepts out of 434,056 possible concepts in the SNOMED-CT and RxNorm subset of UMLS version 2017AB.

[3] https://www.ncbi.nlm.nih.gov/CBBresearch/Dogan/DISEASE/
[4] https://biocreative.bioinformatics.udel.edu/tasks/biocreative-v/
[5] https://sites.google.com/site/shareclefehealth/data
[6] https://n2c2.dbmi.hms.harvard.edu/track3

Dataset	Scientific Articles			Clinical Notes	
	NCBI	BC5CDR-D	BC5CDR-C	ShARe/CLEF	MCN
Ontology	MEDIC	MEDIC	CTD-Chemical	SNOMED-CT	SNOMED-CT & RxNorm
# of Concepts (Ontology)	11,915	11,915	171,203	126,524	434,056
# of Synonyms (Ontology)	71,923	71,923	407,247	520,665	1,550,586
# of Documents (Datasets)	792	1,500	1,500	298	100
# of Concepts (Datasets)	750	1,075	1,164	1,313	3,792
# of Mentions (Datasets)	6,881	12,850	15,935	11,167	13,609

Table 3: Statistics of the five datasets in our experiments.

We take 40 clinical notes from the released data as training, consisting of 5,334 mentions, and the standard evaluation data with 6,925 mentions as our test set. Around 2.7% of mentions in MCN are assigned the *CUI-less* label.

4.2 Implementation details

Unless specifically noted otherwise, we use the same training procedure and hyper-parameter settings across all experiments and on all datasets. As the triplet mining requires at least one positive text in a batch for each anchor text, we randomly sample one positive text for each anchor text and group them into batches. Like previous work (Schroff et al., 2015; Hermans et al., 2017), we adopt euclidean distance to calculate similarity score during training, while at inference time, we compute cosine similarity as it is simpler to interpret. For the sieve-based search, if the cosine similarity score between the mention and the prediction of the first sieve is above 0.95, we use the prediction of first sieve, otherwise, we use the prediction of the second sieve.

When training the triplet network on the combination of the ontology and annotated corpus, we take all the synonyms from the ontology and repeat the concept texts in the annotated corpus such that $\frac{|D|}{|O|} = \frac{1}{3}$. In preliminary experiments we found that large ontologies overwhelmed small annotated corpora. We also experimented with three ratios $\frac{1}{3}$, $\frac{2}{3}$, and 1 between concept texts and synonyms of ontology on NCBI and BC5CDR-D datasets, and found that the ratio of $\frac{1}{3}$ achieves the best performance for Train:OD models. We then kept the same ratio setting for all datasets. We did not thoroughly explore other ratios and leave that to future work.

For all experiments, we use PubMed-BERT (Gu et al., 2020) as the starting point, which pre-trains a BERT-style model from scratch on PubMed abstracts and full texts. In our preliminary experiments, we also tried BioBERT (Lee et al., 2019) as the text encoder, but that resulted in worse performance across five datasets. We use the pytorch implementation of sentence-transformers[7] to train the Triplet Network for concept normalization. We use the following hyper-parameters during the training of the triplet network: sequence_length = 8, batch_size = 1500, epoch_size = 100, optimizer = Adam, learning_rate = 3e-5, warmup_steps = 0.

4.3 Evaluation metrics

The standard evaluation metric for concept normalization is accuracy, because the most similar concept in prediction is of primary interest. For composite mentions like *breast and ovarian cancer* that are mapped to more than one concept in NCBI, BC5CDR-D, and BC5CDR-C datasets, we adopt the evaluation strategy that composite entity is correct if every prediction for each separate mention is correct (Sung et al., 2020).

5 Model selection

We use the development data to choose whether to train the triplet network on just the ontology or also the training data, and to choose which among the similarity search strategies described in section 3.2. Table 4 shows the performance of all such systems across the five different corpora. The top half of the table focuses on settings where the triplet network only needs to be trained once, on the ontology, and the bottom half focuses on settings where the triplet network is retrained for each new dataset. For each half of the table, the last column gives the average of the ranks of each setting's performance across the five corpora. For example, when training the triplet network only on the ontology, the searching strategy D-C (search the training data using concept vectors) is almost always the worst performing,

[7] https://github.com/UKPLab/sentence-transformers

	Train	Search	NCBI	BC5CDR-D	BC5CDR-C	ShARe/CLEF	MCN	Avg. Rank
1	O	O-T	83.74	82.65	97.00	82.76	69.11	10.2
2	O	O-C	85.01	82.43	92.62	81.12	70.96	12
3	O	D-T	85.39	77.29	74.21	79.76	61.26	12.6
4	O	D-C	85.26	75.18	74.11	69.70	59.70	13.6
5	O	OD-T	89.58	88.87	97.75	88.12	72.67	4.8
6	O	OD-C	88.56	85.85	93.30	82.23	72.59	9.4
7	O	D-T + O-T	90.34	89.66	97.62	87.26	81.33	3.6
8	O	D-T + O-C	89.96	89.40	96.88	83.73	81.93	5
9	O	D-C + O-T	86.28	83.72	97.14	82.98	76.67	7.4
10	O	D-C + O-C	88.56	83.51	95.77	81.58	76.52	9.8
11	O	D-T + OD-T	91.36	**90.50**	97.64	**90.50**	81.85	2
12	O	D-T + OD-C	90.85	89.90	96.88	84.69	**82.15**	3.6
13	O	D-C + OD-T	**91.99**	89.47	**97.76**	86.83	79.19	3.2
14	O	D-C + OD-C	88.82	86.93	96.32	82.55	77.41	7.6
15	OD	O-T	89.58	87.82	96.71	86.62	72.37	9.8
16	OD	O-C	91.36	89.85	96.32	88.11	80.52	9.6
17	OD	D-T	86.40	79.01	74.23	79.87	63.33	13.2
18	OD	D-C	86.40	78.41	74.23	80.19	62.52	13.4
19	OD	OD-T	91.11	90.38	97.85	88.87	76.15	8.2
20	OD	OD-C	91.61	89.92	96.32	88.33	81.4	7.8
21	OD	D-T + O-T	91.25	91.10	97.81	90.15	84.37	4
22	OD	D-T + O-C	91.49	90.88	96.22	88.76	84.52	6.4
23	OD	D-C + O-T	92.25	90.71	97.87	89.61	83.78	4
24	OD	D-C + O-C	91.49	90.47	96.28	88.65	83.93	7.8
25	OD	D-T + OD-T	91.61	**91.22**	97.81	**90.21**	84.37	2.4
26	OD	D-T + OD-C	91.61	90.83	96.22	89.08	**84.67**	5.2
27	OD	D-C + OD-T	**92.25**	90.95	**97.91**	90.15	83.70	3.4
28	OD	D-C + OD-C	91.61	90.55	96.28	89.40	84.00	5.8

Table 4: Dev performances of the triplet network trained on ontology and ontology + data with different similarity search strategies. The last column *Avg. Rank* shows the average rank of each similarity search strategy across multiple datasets. Models with best average rank are highlighted in grey; models with best accuracy are bolded.

ranking 14th of 14 in four corpora and 12th of 14 in one corpus, for an average rank of 13.6.

Table 4 shows that the best models search over both the ontology and the training data. Models that only search over the training data (D-T and D-C) perform worst, with average ranks of 12.6 or higher regardless of what the triplet network is trained on, most likely because the training data covers only a fraction of the concepts in the test data. Models that only search over the ontology (O-T and O-C) are only slightly better, with average ranks between 9.6 and 12, though the models in the first two rows of the table at least have the advantage that they require no annotated training data (they train on and search over only the ontology). However, the performance of such models can be improved by adding domain-specific synonyms to the ontology, i.e., OD-T vs. O-T (rows 5 vs. 1), and OD-C vs. O-C (rows 6 vs. 2), or adding domain-specific synonyms and then searching in a sieve-based manner (rows 7-14).

Table 4 also shows that searching based on text (ontology synonyms or training data mentions) vectors typically outperforms searching based on concept (average of text) vectors. Each pair of rows in the table shows such a comparison, and only in rows 15-16 and 19-20 are the average ranks of the -C models higher than the -T models.

Table 4 also shows that models using mixed representation types (-T and -C) have worse ranks than the text-only models (-T). For instance, going from Train:O-Search:O-C to Train:O-Search:O-T improves the average rank from 12 to 10.2, going from Train:OD-Search:D-T+OD-C to Train:OD-Search:D-T+OD-T improves the average rank from 5.2 to 2.4, etc. There are a few exceptions to this on the MCN dataset. We analyzed the differences in the predictions of Train:OD-Search:D-T+OD-T (row 25) and Train:OD-Search:D-T+OD-C (row 26) on this dataset, and found that concept vectors sometimes helps to solve ambiguous mentions by averaging their concept texts. For instance, the OD-T model finds concepts *C0013144* and *C2830004* for mention *somnolent* as they have the overlapping synonym *somnolent*, while the OD-C model ranks *C2830004* higher as the other concept also has other synonyms such as *Drowsy*, *Sleepiness*.

Finally, table 4 shows that sieve-based models

Approach	NCBI	BC5CDR-D	BC5CDR-C	ShARe/CLEF	MCN
Sieve-based (D'Souza and Ng, 2015)	84.65	-	-	90.75	-
Sieve-based (Luo et al., 2019)	-	-	-	-	76.35
TaggerOne (Leaman and Lu, 2016)	88.80	88.9	94.1	-	-
CNN-based ranking (Li et al., 2017)	86.10	-	-	90.30	-
BERT-based ranking (Ji et al., 2020)	89.06	-	-	**91.10**	-
BERT-based ranking (Xu et al., 2020)	-	-	-	-	83.56
BIOSYN (Sung et al., 2020)	91.1	**93.2**	96.6	-	-
TTI (Henry et al., 2020)	-	-	-	-	**85.26**
PubMed-BERT + Search:O-T	76.56	76.60	91.78	73.64	59.97
PubMed-BERT + Search:D-T+OD-T	82.19	90.53	94.24	85.35	75.81
Train:O + Search:O-T	82.60	84.44	95.79	83.48	69.62
Train:O + Search:D-T+OD-T	89.48	92.30	96.67	89.19	82.19
Train:OD + Search:D-T+OD-T	88.96	92.92	96.81	90.41	83.23
Train:OD + Search:tuned	**91.15**	92.92	**96.91**	90.41	83.70

Table 5: Comparisons of our proposed approaches against the current state-of-the-art performances on *NCBI*, *BC5CDR-D*, *BC5CDR-C*, *ShARe/CLEF*, and *MCN* datasets. Approaches with best accuracy are bolded.

outperform their non-sieve-based counterparts. For example, D-T + O-T has better average ranks than O-T, D-T, or OD-T (rows 7 vs. 1, 3, and 5; and rows 21 vs. 15, 17, and 19).

From this analysis on the dev set, we select the following models to evaluate on the test set:

Train:O + Search:O-T This is the best approach that requires only the ontology; no annotated training data is used.

Train:O + Search:D-T+OD-T This is the best approach that only needs to be trained once (on the ontology), as the training data is only used to add extra concept text during search time. This is similar to a real-world scenario where a user manually adds some extra domain-specific synonyms for concepts they care about.

Train:OD + Search:D-T+OD-T This is the best approach that can be created from any combination of ontology and training data. The triplet network must be retrained for each new domain.

Train:OD + Search:tuned This is the bold models in the second half of table 4. It requires not only retraining the triplet network for each new domain, but also trying out all search strategies on the new domain and selecting the best one.

6 Results

Table 5 shows the results of our selected models on the test set, alongside the best models in the literature. Our Train:OD+Search:tuned model achieves new state-of-the-art on BC5CDR-C (p[8]=0.0291), equivalent performance on NCBI (p=0.6753) and BC5CDR-D (p=0.1204), <1 point worse on ShARe (p=0.0375), and <2 points worse on MCN (p=0). Note that the performance of TTI is from an ensemble of multiple system runs. Yet this model is simpler than most prior work: it requires no two-step generate-and-rank framework (Li et al., 2017; Ji et al., 2020; Xu et al., 2020), no iterative candidate retrieval over the entire training data (Sung et al., 2020), no hand-crafted rules or features (D'Souza and Ng, 2015; Leaman and Lu, 2016; Luo et al., 2019), and no acronym expansion or TF-IDF transformations (D'Souza and Ng, 2015; Ji et al., 2020; Sung et al., 2020).

The PubMed-BERT rows in Table 5 demonstrate that the triplet training is a critical part of the success: if we use PubMed-BERT without triplet training, performance is 2 to 8 points worse than our best models, depending on the dataset. Yet, we can see that our proposed search strategies are also important, as on the BC5CDR datasets, PubMed-BERT can get within 3 points of the state-of-the-art using the D-T+OD-T search strategy (though it is much further away on the other datasets).

Perhaps most interestingly, our triplet network trained only on the ontology and no annotated training data, Train:O+Search:D-T+OD-T, achieves within 3 points of state-of-the-art on all datasets. We believe this represents a more realistic scenario: unlike prior work, our triplet network does not need to be retrained for each new dataset/domain if their concepts are from the same ontology. Instead, the model can be adapted to a new dataset/domain by simply pointing out any extra domain-specific synonyms for concepts, and the search can integrate these directly. Domain-specific synonyms do

[8]We used a one-sample bootstrap resampling test. The one sample is 10,000 runs of bootstrapping results of our system.

	PubMed-BERT + Search:OD-T			Train:O + Search:OD-T			Train:OD + Search:OD-T		
Rank	Text	Concept	Score	Text	Concept	Score	Text	Concept	Score
1	HNSCC	C535575	0.919	Hyperparathyroidism, Primary	D049950	0.767	Hyperparathyroidism, Primary	D049950	0.838
5	NPC2	C536119	0.903	Hyperparathyroidism 1	C564166	0.692	Primary Hyperparathyroidism	D049950	0.830
10	MPNST	D009442	0.900	HRPT1	C564166	0.611	HRPT1	C564166	0.672
15	HPNS	D006610	0.897	Hyperparathyroidism 2	C563273	0.595	Parathyroid Adenoma, Familial	C564166	0.644
20	PBC2	C567817	0.895	Hyperparathyroidism, Secondary	D006962	0.566	Hyperparathyroidisms, Secondary	D006962	0.608

Table 6: Top similar texts, their concepts, and similarity scores for mention *primary HPT* (*D049950*) predicted from models PubMed-BERT + Search:OD-T, Train:O + Search:OD-T and Train:OD + Search:OD-T.

seem to be necessary for all datasets; without them (i.e., Train:O+Search:O-T), performance is about 10 points below state-of-the-art.

As a small qualitative analysis of the models, Table 6 shows an example of similarity search results, where the systems have been asked to normalize the mention *primary HPT*. PubMed-BERT fails, producing unrelated acronyms, while both triplet network models find the concept and rank it with the highest similarity score.

7 Limitations and future research

Our ability to normalize polysemous concept mentions is limited by their context-independent representations. Although our PubMed-BERT encoder is a pre-trained contextual model, we feed in only the mention text, not any context, when producing a representation vector. This is not ideal for mentions with multiple meanings, e.g., *potassium* in clinical notes may refer to the substance (C0032821) or the measurement (C0202194), and only the context will reveal which one. A better strategy to generate the contextualized representation for the concept mention, e.g., Schumacher et al. (2020), may yield improvements for such mentions.

We currently train a separate triplet network for each ontology (one for MEDIC, one for CTD, one for SNOMED-CT, etc.) but in the future we would like to train on a comprehensive ontology like the UMLS Metathesaurus (Bodenreider, 2004), which includes nearly 200 different vocabularies (SNOMED-CT, MedDRA, RxNorm, etc.), and more than 3.5 million concepts. We expect such a general vector space model would be more broadly useful to the biomedical NLP community.

We explored one type of triplet training network, but in the future we would like to explore other variants, such as semi-hard triplet mining (Schroff et al., 2015) for generating samples, cosine similarity for measuring the similarity during training and inference, and multi-similarity loss (Wang et al., 2019) for calculating the loss.

8 Conclusions

We presented a vector-space framework for concept normalization, based on pre-trained transformers, a triplet objective with online hard triplet mining, and a new approach to vector similarity search. Across five datasets, our models that require only an ontology to train are competitive with state-of-the-art models that require domain-specific training.

Acknowledgements

Research reported in this publication was supported by the National Library of Medicine and the National Institute of General Medical Sciences of the National Institutes of Health under Award Numbers R01LM012918 and R01GM114355. The computations were done in systems supported by the National Science Foundation under Grant No. 1228509. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

References

Olivier Bodenreider. 2004. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32(suppl_1):D267–D270.

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*, 2012.

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.

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.

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 (Volume 2: Short Papers)*, pages 297–302, Beijing, China. Association for Computational Linguistics.

Albert Gordo, Jon Almazán, Jerome Revaud, and Diane Larlus. 2016. Deep image retrieval: Learning global representations for image search. In *European conference on computer vision*, pages 241–257. Springer.

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 preprint arXiv:2007.15779*.

Sam Henry, Yanshan Wang, Feichen Shen, and Ozlem Uzuner. 2020. The 2019 National Natural language processing (NLP) Clinical Challenges (n2c2)/Open Health NLP (OHNLP) shared task on clinical concept normalization for clinical records. *Journal of the American Medical Informatics Association*, 27(10):1529–1537.

Alexander Hermans, Lucas Beyer, and Bastian Leibe. 2017. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*.

Elad Hoffer and Nir Ailon. 2015. Deep metric learning using triplet network. In *International Workshop on Similarity-Based Pattern Recognition*, pages 84–92. Springer.

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

Jitendra Jonnagaddala, Toni Rose Jue, Nai-Wen Chang, and Hong-Jie Dai. 2016. Improving the dictionary lookup approach for disease normalization using enhanced dictionary and query expansion. *Database*, 2016:baw112.

Ning Kang, Bharat Singh, Zubair Afzal, Erik M van Mulligen, and Jan A Kors. 2013. Using rule-based natural language processing to improve disease normalization in biomedical text. *Journal of the American Medical Informatics Association*, 20(5):876–881.

Rohit J. Kate. 2016. Normalizing clinical terms using learned edit distance patterns. *Journal of the American Medical Informatics Association*, 23(2):380–386.

André Leal, Bruno Martins, and Francisco Couto. 2015. ULisboa: Recognition and normalization of medical concepts. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 406–411, Denver, Colorado. Association for Computational Linguistics.

Robert Leaman, Rezarta Islamaj Doğan, and Zhiyong Lu. 2013. DNorm: disease name normalization with pairwise learning to rank. *Bioinformatics*, 29(22):2909–2917.

Robert Leaman and Zhiyong Lu. 2016. Taggerone: joint named entity recognition and normalization with semi-markov models. *Bioinformatics*, 32(18):2839–2846.

Robert Leaman, Chih-Hsuan Wei, and Zhiyong Lu. 2015. tmchem: a high performance approach for chemical named entity recognition and normalization. *Journal of cheminformatics*, 7(S1):S3.

Hsin-Chun Lee, Yi-Yu Hsu, and Hung-Yu Kao. 2016. AuDis: an automatic CRF-enhanced disease normalization in biomedical text. *Database*, 2016. Baw091.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*. Btz682.

Kathy Lee, Sadid A. Hasan, Oladimeji Farri, Alok Choudhary, and Ankit Agrawal. 2017. Medical Concept Normalization for Online User-Generated Texts. In *2017 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 462–469. IEEE.

Fei Li, Yonghao Jin, Weisong Liu, Bhanu Pratap Singh Rawat, Pengshan Cai, and Hong Yu. 2019. Fine-Tuning Bidirectional Encoder Representations From Transformers (BERT)–Based Models on Large-Scale Electronic Health Record Notes: An Empirical Study. *JMIR Med Inform*, 7(3):e14830.

Haodi Li, Qingcai Chen, Buzhou Tang, Xiaolong Wang, Hua Xu, Baohua Wang, and Dong Huang. 2017. Cnn-based ranking for biomedical entity normalization. *BMC bioinformatics*, 18(11):79–86.

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. 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 (Volume 1: Long Papers)*, pages 1014–1023, Berlin, Germany. Association for Computational Linguistics.

Hongwei Liu and Yun Xu. 2017. A Deep Learning Way for Disease Name Representation and Normalization. In *Natural Language Processing and Chinese Computing*, pages 151–157. Springer International Publishing.

Yen-Fu Luo, Weiyi Sun, and Anna Rumshisky. 2019. MCN: A comprehensive corpus for medical concept normalization. *Journal of Biomedical Informatics*, pages 103–132.

Xiaofei Ma, Peng Xu, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. 2019. Universal text representation from bert: An empirical study. *arXiv preprint arXiv:1910.07973*.

Zulfat Miftahutdinov and Elena Tutubalina. 2019. Deep neural models for medical concept normalization in user-generated texts. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 393–399, Florence, Italy. Association for Computational Linguistics.

Ishani Mondal, Sukannya Purkayastha, Sudeshna Sarkar, Pawan Goyal, Jitesh Pillai, Amitava Bhattacharyya, and Mahanandeeshwar Gattu. 2019. Medical entity linking using triplet network. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 95–100, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Shikhar Murty, Patrick Verga, Luke Vilnis, Irena Radovanovic, and Andrew McCallum. 2018. Hierarchical losses and new resources for fine-grained entity typing and linking. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 97–109, Melbourne, Australia. Association for Computational Linguistics.

Paul Neculoiu, Maarten Versteegh, and Mihai Rotaru. 2016. Learning text similarity with Siamese recurrent networks. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 148–157, Berlin, Germany. Association for Computational Linguistics.

Thanh Ngan Nguyen, Minh Trang Nguyen, and Thanh Hai Dang. 2018. Disease Named Entity Normalization Using Pairwise Learning To Rank and Deep Learning. Technical report, VNU University of Engineering and Technology.

Jinghao Niu, Yehui Yang, Siheng Zhang, Zhengya Sun, and Wensheng Zhang. 2019. Multi-task Character-Level Attentional Networks for Medical Concept Normalization. *Neural Process Lett*, 49(3):1239–1256.

Nikhil Pattisapu, Sangameshwar Patil, Girish Palshikar, and Vasudeva Varma. 2020. Medical Concept Normalization by Encoding Target Knowledge. In *Proceedings of the Machine Learning for Health NeurIPS Workshop*, volume 116 of *Proceedings of Machine Learning Research*, pages 246–259. PMLR.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

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, Online. Association for Computational Linguistics.

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*, pages 3641–3650, Online. Association for Computational Linguistics.

Hanna Suominen, Sanna Salanterä, Sumithra Velupillai, Wendy W Chapman, Guergana Savova, Noemie Elhadad, Sameer Pradhan, Brett R South, Danielle L Mowery, Gareth JF Jones, et al. 2013. Overview of the share/clef ehealth evaluation lab 2013. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 212–231. Springer.

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.

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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Chih-Hsuan Wei, Yifan Peng, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Jiao Li, Thomas C Wiegers, and Zhiyong Lu. 2015.

Overview of the biocreative v chemical disease relation (cdr) task. In *Proceedings of the fifth BioCreative challenge evaluation workshop*, volume 14.

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, Online. Association for Computational Linguistics.