Robust Representation Learning of Biomedical Names

Minh C. Phan **Aixin Sun** **Yi Tay**
Nanyang Technological University, Singapore
phan0050@e.ntu.edu.sg; axsun@ntu.edu.sg; ytay017@e.ntu.edu.sg

Abstract

Biomedical concepts are often mentioned in medical documents under different name variations (synonyms). This mismatch between surface forms is problematic, resulting in difficulties pertaining to learning effective representations. Consequently, this has tremendous implications such as rendering downstream applications inefficacious and/or potentially unreliable. This paper proposes a new framework for learning robust representations of biomedical names and terms. The idea behind our approach is to consider and encode contextual meaning, conceptual meaning, and the similarity between synonyms during the representation learning process. Via extensive experiments, we show that our proposed method outperforms other baselines on a battery of retrieval, similarity and relatedness benchmarks. Moreover, our proposed method is also able to compute meaningful representations for unseen names, resulting in high practical utility in real-world applications.

1 Introduction

Representation learning of words (Mikolov et al., 2013; Pennington et al., 2014), and/or sentences (Kiros et al., 2015; Hill et al., 2016; Logeswaran and Lee, 2018) forms the bedrock of many modern NLP applications. These techniques, largely relying on context information, have a huge impact on downstream applications. To this end, learning effective and useful representations has been a highly fruitful area of research.

Biomedical names[1], however, are different from standard words and sentences. These names have both contextual and conceptual meanings. Contextual meaning reflects the contexts where the names appear, and it is specifically granted to each name. Names of a broad and popular concept often have slightly different contextual meanings. On the other hand, conceptual meaning maps to the definitions/contexts of the names' associated concepts, *i.e.,* CUIs as shown in Table 1. As such, names of the same concepts share the common conceptual meanings, although they can own different contextual information.

Concept (CUI) and their names	Source
C0343047: leiner's disease, complement component 5 deficiency, c5d, complement 5 dysfunction, infantile seborrheic dermatitis, erythroderma desquamativum.	UMLS
C0154832: coats' disease, abnormal retinal vascular development, unilateral retinal telangiectasis, coats telangiectasis	NCBI-Disease
C0019168: hepatitis b virus surface antigen, hepatitis-b surface antigen, hbs ag, hbsag, hepatitis b surface antigen	BC5CDR-Chemical

Table 1: Example of biomedical concepts and their names taken from one vocabulary (UMLS (Li et al., 2016)) and two annotated datasets (NCBI-Disease (Doğan et al., 2014) and BC5CDR-Chemical-Chemical (Li et al., 2016)). The concepts are listed by concept unique identifiers (CUI) defined in UMLS.

As illustrated in Table 1, biomedical concepts appear in the text under various names. Representations of the names are also expected to be well clustered in their distributional space, *i.e.,* names of the same concepts are close to each other and distant from those of other concepts. Learning such conceptually grounded representations is highly desired for a wide range of applications, e.g., synonym retrieval/discovery, biomedical name normalization, and query expansion.

For the first time, we investigate the problem of biomedical name embedding. Our goal is to derive meaningful and robust representations for biomedical names from their surface forms. Unfortunately, this task is not trivial since two names

[1]Biomedical names refer to surface forms that represent biomedical concepts. They can be official names in biomedical vocabularies or unofficial names mentioned in text.

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3275–3285
Florence, Italy, July 28 - August 2, 2019.

can be strongly related but not necessarily belong to the same concept (*e.g.,* 'complement component 5 deficiency' and 'complement component 5'). Furthermore, names of a concept can be completely different regarding their surface forms (*e.g.,* 'leiner's disease' and 'c5d'). As such, we establish the key desiderata for learning robust representations. First, the output representations need to be both conceptually and contextually meaningful. Second, name representations that belong to the same concepts should be similar to each other, *i.e.,* conceptual grounding.

To this end, our proposed encoding framework incorporates three new objectives, namely context, concept, and synonym-based objectives. We formulate the representation learning process as a synonym prediction task, with context and conceptual losses acting as regularizers, preventing two synonyms from collapsing into semantically meaningless representations. As illustrated in Figure 1, synonym-based objective enforces similar representations between synonymous names, while concept-based objective pulls the name's representations closer to its concept's centroid. On the other hand, context-based objective aims to minimize the difference between the derived representation and its specific contextual representation. More concretely, our approach adopts a recurrent sequence encoding model to extract the semantics of biomedical names, and to learn the alternative naming of biomedical concepts. Our approach does not need any additional annotations on biomedical text. To be specific, we do not need the biomedical names to be pre-annotated in the text. Instead, we utilize available synonym sets in a metathesaurus vocabulary (*e.g.,* UMLS), as the only additional resource for training.

Our main contributions in this work are summarized as follows. For the first time, we investigate the problem of biomedical name embedding and its applications. We pay attention to the similarity between semantically related names as well as the names of the same concept. Furthermore, we define and distinguish three aspects constituting to quality of biomedical name representations. We propose a novel encoding framework that considers all these aspects in the representation learning. Finally, we evaluate the proposed encoder in biomedical synonym retrieval, name normalization, and semantic similarity and relatedness benchmarks. In most of these experiments, our model significantly outperforms other baselines.

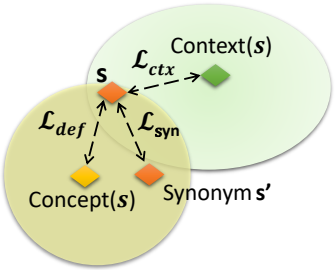


Figure 1: Illustration of three aspects, which are associated to three training objectives, for computing representation of biomedical name s. Intuitively, the representation is supposed to be similar to its synonym's as well as its conceptual and contextual representations.

2 Related Work

Our problem setting of name embedding is different from recent works in biomedical word embeddings (Chiu et al., 2016; Wang et al., 2018) and concept embeddings (Beam et al., 2018; Cai et al., 2018). Our goal is to derive meaningful representation for a sequence of words that likely represents a concept. This setting is also orthogonal to works that only focus on estimating the matching between names (Li et al., 2017; Liu et al., 2018).

There are several options to encode variable-length names/phrases into fixed-sized vector representations. Existing approaches range from phrase-level extensions of word embeddings, compositions of pre-trained word representations to sequence encoding neural networks.

Contextual Word Embeddings. We revisit skip-gram model (Mikolov et al., 2013), as one of the most popular context-based embedding approaches. The model computes the representations for both target word w_t, and context word w_c by maximizing the following log-likelihood:

$$\mathcal{L}_W = \sum_{w_t, w_c \in \mathcal{C}_{w_t}} log\, p(w_c|w_t) \qquad (1)$$

The probability of observing w_c in the local context of w_t is defined as follows:

$$p(w_c|w_t) = \frac{exp(v_{w_c}^{\intercal} u_{w_t})}{\sum_{w \in \mathcal{W}} exp(v_w^{\intercal} u_{w_t})}$$

where u_w and v_w are the 'input' and 'output' vector representations of w. In this work, we refer to the input representations as contextual representations of words, or in short, word embeddings.

The skip-gram model is extensible to names (or phrases) by treating them as special tokens:

$$\mathcal{L}_S = \sum_{w_t, w_c \in \mathcal{C}_{w_t}} log\, p(w_c|w_t) + \sum_{s, w_c \in \mathcal{C}_s} log\, p(w_c|s) \quad (2)$$

where s is a special name token. Training of this model results in word and name embeddings.

Average of Contextual Word Embeddings. Another simple and effective method to compute name embeddings is taking the average of their constituent word embeddings. Since words in a biomedical name are usually descriptive about its meaning, this simple baseline is expected to produce quality representations. FastText (Bojanowski et al., 2017) leverages this idea by considering character n-grams instead of words. Therefore, the model can derive representations for names that contain unseen words. The effectiveness of simple compositions such as taking average or power mean have also been verified in phrase and sentence embeddings (Wieting et al., 2016; Arora et al., 2017; Rücklé et al., 2018).

Sequence Encoding Models. Sequence encoding models aim to capture more sophisticated semantics of character and word sequences. These models range from multilayer feed-forward networks (Iyyer et al., 2015) to convolutional (Kalchbrenner et al., 2014), recursive and recurrent neural networks (Socher et al., 2011; Tai et al., 2015). They also differ by the types of supervision used in training. Context-based sentence encoders (Kiros et al., 2015; Hill et al., 2016; Logeswaran and Lee, 2018) is based on distributional hypothesis. The training utilizes sentences and their contexts (surrounding sentences), which can be extracted from an unlabeled corpus. Similar to contextual word embeddings, the derived sentence embeddings are expected to carry the contextual information. However, this contextual information does not fully reflect paraphrastic characteristic, *i.e.,* semantically similar sentences do not necessarily have identical meanings. These embeddings, therefore, are not favorable in applications that demand strong synonym identification. In contrast, supervised or semi-supervised representation leaning requires annotated corpus, such as paraphrastic sentences or natural language inference data (Conneau et al., 2017; Wieting and Gimpel, 2017; Clark et al., 2018; Subramanian et al., 2018; Cer et al., 2018). However, most of these works focus on learning representations for sentences.

The closest work to our problem setting is (Wieting et al., 2015). In this proposed model, the authors utilize pairs of paraphrastic phrases as training data, *e.g.,* 'does not exceed' and 'is no more than'. To prevent the trained model from overfitting, authors introduce regularization terms that applied on encoder's parameters as well as the difference between the initial and trainable word embeddings. Their evaluation, however, only considers the paraphrastic similarity of phrases.

Discussion. Our proposed encoder is based on BiLSTM (Graves and Schmidhuber, 2005), although it can be replaced by another sequence encoding model as mentioned above. Our approach utilizes synonym sets in UMLS to learn name representations, while also enforces the learned representation to be similar to their contextual and conceptual representations. The idea is related to word vector specialization (retrofitting) (Faruqui et al., 2015; Mrkšić et al., 2017; Vulić et al., 2018). The difference is that we focus on learning representation for multi-word concept names, hence the contextual and conceptual constraints are essential, in addition to the synonymous similarity. In contrast, most retrofitting approaches mainly aim to improve word representations. These models map initial word embeddings into a new vector space that satisfy the synonymous similarity desiderata, while also constrain the new representations to be similar to the initial ones. Since the initial word representations can be assumed to encode both contextual and conceptual information of the words, these retrofitting approaches can be viewed as special cases of our proposed encoding framework.

3 Biomedical Name Encoder

For ease of presentation, we use three generic terms, u_w, u_s and u_c, to denote pre-trained word, name and concept embeddings, respectively. These embeddings will be used as inputs in our encoding framework. Note that there are several options to calculate these embeddings and our encoder can be adapted to different calculation results. Before going to details, we present an extension of skip-gram, which will serve as a baseline. Furthermore, the outputs of this baseline will be used as pre-trained embeddings in one of the framework's configurations.

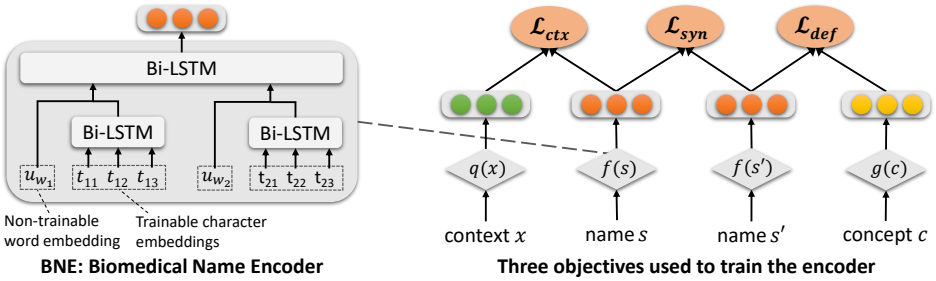


Figure 2: Our proposed biomedical name encoding framework. The main encoder (BNE) is based on two-level BiLSTM to capture both character and word-level information of an input name. BNE parameters are learned by considering three training objectives. Synonym-based objective $\mathcal{L}_{syn}$ enforces similar representations between two synonymous names (s and s'). Concept-based objective $\mathcal{L}_{def}$, and context-based objectives $\mathcal{L}_{ctx}$ apply similarity constraints on representations of names (s or s', which are interchangeable) and their conceptual and contextual representations ($g(c)$ and $g(x)$, respectively). Details about $g(c)$ and $g(x)$ calculations are discussed in Section 3.2.

3.1 Skip-gram with Context and Concept

The skip-gram model described by Equation 2 uses context words to calculate embeddings for names. Apart from the context words, we also considers the name's conceptual information in this new baseline. We leverage two sources of conceptual information: words in a name, and name's associated concept. We assume that names containing similar words tend to have similar meaning. Furthermore, names of the same concepts will also share common meaning.

We introduce a new token type for concepts. The concept embeddings are trained in a similar way as name embeddings. Specifically, for this baseline, we utilize a pre-annotated corpus where names appearing in the training text are labeled with their associated concepts. We convert the annotated texts into sequences of words, name, and concept tokens to be used as inputs to the skip-gram model. For example, consider a pseudo sentence that has 4 words and contains a bigram name: w_l $\underline{w_1\ w_2}$ w_r, we map the annotated name $w_1\ w_2$ to a name token s_i, and its annotated concept is denoted by c_i. We create two sequences of tokens corresponding to this original sentence:

- $w_l,\ s_i,\ c_i,\ w_1,\ w_2,\ w_r$
- $w_l,\ w_1,\ w_2,\ s_i,\ c_i,\ w_r$

The name and concept tokens are placed on the left and right sides of the annotated name to avoid being biased toward any single side. These token sequences are subsequently fed as inputs to the skip-gram baseline (the training details are presented in Section 4). Outputs of this baseline are word, name and concept embeddings.

3.2 Biomedical Name Encoder with Context, Concept, and Synonym

Our proposed framework is illustrated in Figure 2. The encoder unit is based on BiLSTM to aggregate information from both character and word levels. The encoded representations are constrained by three objectives, namely synonym, context, and concept-based objectives. The model utilizes synonym sets in UMLS as training data. We denote all the synonym sets as $\mathcal{U} = \{\mathcal{S}_c\}$, where $\mathcal{S}_c$ includes all names of concept c, *i.e.*, $\mathcal{S}_c = \{s_i\}$.

Biomedical Name Encoder (BNE). The encoder extracts a fixed-sized representation for a given name (or surface form) s. We use one BiLSTM unit with last-pooling to encode character-level information of each word. The representation is then concatenated with the pre-trained word embedding to form a word-level representation. Another BiLSTM unit with max-pooling is used to aggregate the semantics from the sequence of words' representations. Finally the aggregated representation is passed through a linear transformation. Mathematically, the encoding function is expressed as follows:

$$h_{w_i} = [u_{w_i} \oplus last(BiLSTM(t_{i,1}, .., t_{i,m}))]$$
$$h_s = max(BiLSTM(h_{w_1}, .., h_{w_n}))$$
$$f(s) = Wh_s + b$$

where u_{w_i} represents the pre-trained word embedding of word w_i in name s. $t_{i,j}$ is a trainable character embedding in w_i. $\oplus$ denotes vector concatenation. W and b are parameters of the last transformation. Next, we detail three objectives used to train the encoder.

Synonym-based Similarity. Representations of names that belong to the same concept should be similar to each other. We formulate this objective using the following loss function:

$$\mathcal{L}_{syn} = \sum_{(s,s') \in \mathcal{S}_c \times \mathcal{S}_c} d(f(s), f(s')) \quad (3)$$

where $d(\cdot, \cdot)$ is a function that measures the difference between two representations.

As mentioned in the introduction, training the encoder using only this synonym-based objective will lead to biased representations. Specifically, the encoder will be trained to act like a hash function, which performs well on determining whether two names are synonym of each other. However, it likely loses the semantics of names. As a remedy, we further introduce concept and context-based objectives to regularize the representations.

Conceptual Meaningfulness. Representations of biomedical names should be similar to those of their associated concepts. This objective complements the synonym-based objective introduced earlier. The latter not only shifts the synonymous embeddings close to each other, but also pulls them near to its concept's centroid, expressed as:

$$\mathcal{L}_{def} = \sum_{c,\ s \in \mathcal{S}_c} d(f(s), g(c)) \quad (4)$$

where $g(c)$ returns a vector that encodes conceptual information of the corresponding concept c. There are several options for this representation. It can be a mapping to pre-trained concept embeddings learned from a large corpus, *i.e.,* $g(c) = u_c$. Another option is taking composition (*e.g.,* average) of all its name embeddings (see Table 1), *i.e.,* $g(c) = \frac{1}{|\mathcal{S}_c|} \sum_{s \in \mathcal{S}_c} u_s$. Furthermore, when definition of the concept is available, $g(c)$ can be modeled as another encoding function that extracts the conceptual meaning from the definition.

Contextual Meaningfulness. Each name representation should accommodate specific contextual information owned by the name, formulated as:

$$\mathcal{L}_{ctx} = \sum_{s, x \in \mathcal{X}_s} d(f(s), q(x)) \quad (5)$$

where $\mathcal{X}_s$ represents all local contexts of name s, and $q(x)$ returns contextual representation of local context x. A straightforward way to model $\mathcal{X}_s$ is using local context words of s. However, this modeling is computationally expensive since the training will need to iterate through all the context words of the name. Alternatively, the contextual information can be modeled using 1-hop approximation of the name's local contexts, which is mapped to the name's contextual representation, *i.e.,* $\mathcal{X}_s = \{s\}$ and $q(x) = q(s) = u_s$. We also consider another approximation where the contextual representation is further approximated by its pre-trained word embeddings, *i.e.,* $q(s) = \frac{1}{|\mathcal{T}(s)|} \sum_{w \in \mathcal{T}(s)} u_w$ where $\mathcal{T}(s)$ represents words in name s. Intuitively, in these two approximations, we assume that the pre-trained name or word embeddings carry local contextual information since they are trained by context-based approaches (see Section 2).

Combined Loss Function. The final loss function combines all the introduced losses:

$$\mathcal{L}_{BNE} = \mathcal{L}_{syn} + \mathcal{L}_{def} + \mathcal{L}_{ctx} \quad (6)$$

For simplicity, we ignore weighting factors that control the contribution of each loss. However, applying and fine-tuning these factors will shift the encoding results more on either semantic similarity or synonym-based similarity direction.

Choices of $g(c)$ and $q(x)$. Several options to calculate the conceptual and contextual representations are discussed earlier. Note that the two representations should be placed in the same distributional space. As such, the implicit relations between them are encoded in, and can be decoded from, their presentations. For efficiency, we model the local contexts $\mathcal{X}_s$ using contextual information encoded in the name itself, *i.e.,* $\mathcal{X}_s = \{s\}$ and $q(x) = q(s)$. To this end, we focus on studying two combinations of $g(c)$ and $q(s)$:

- Option 1: Both $g(c)$ and $q(s)$ directly map to the pre-trained concept and name embeddings, respectively, *i.e.,* $g(c) = u_c$ and $q(s) = u_s$. These embeddings are the outputs of our proposed extension of skip-gram model (see Section 3.1). This option requires annotated biomedical corpus.

- Option 2: The contextual presentation $q(s)$ is approximated by the average of pre-trained words embeddings, *i.e.,* $q(s) = \frac{1}{|\mathcal{T}(s)|} \sum_{w \in \mathcal{T}_s} u_w$; and $g(c)$ is the average of all contextual presentations associated to the

concept, *i.e.,* $g(c) = \frac{1}{|\mathcal{S}_c|}\sum_{s\in\mathcal{S}_c} q(s)$. These computations only require pre-trained word embeddings, and a dictionary of names and concepts, *e.g.,* UMLS.

Distance Function and Optimization. Distance function d can be Euclidean distance or Kullback-Leibler divergence. Alternatively, the optimization can be modeled as binary classification, motivated by its efficiency and effectiveness (Conneau et al., 2017; Wieting and Gimpel, 2017; Logeswaran and Lee, 2018). Another benefit of using classification is to align the encoded BNE vectors to the pre-trained word, name, and concept embeddings. The pre-trained embeddings are derived by skip-gram with negative sampling (Mikolov et al., 2013), which is also formulated as classification. In a similar way, we adopt logistic loss with dot product classifier for all the objectives. For example, the updated loss function for $\mathcal{L}_{syn}$ is rewritten as follows:

$$\ell(f(s')^\intercal f(s)) + \sum_{\bar{s}\in\mathcal{N}_s} \ell(-f(\bar{s})^\intercal f(s))$$

where ℓ is the logistic loss function $\ell : x \mapsto log(1 + e^{-x})$. Negative name $\bar{s}$ is sampled from a mini-batch during optimization, similar to (Wieting et al., 2015). In a similar way, the loss functions $\mathcal{L}_{def}$ and $\mathcal{L}_{ctx}$ are also updated accordingly.

4 Experiments

We first detail the implementations of baselines and the proposed BNE model. We then evaluates all the models with 4 different tasks in retrieval, embedding similarity and relatedness benchmarks.

Skip-gram Baselines. We consider three variants of skip-gram (with negative sampling). **SG$_{\mathbf{W}}$** obtains word embeddings by training the very basic skip-gram model (see Equation 1). To get the representation for a name, we simply take the average of its associated word embeddings. **SG$_{\mathbf{S}}$** is another variant that considers names as special tokens. The model obtains embeddings for word and names concurrently (see Equation 2). **SG$_{\mathbf{S}}$** training requires input text to be segmented into names and regular words. **SG$_{\mathbf{S.C}}$** is our proposed extension of skip-gram model. As introduced in Section 3.1, this baseline requires an annotated corpus where the names are labeled with their associated concepts.

Training of Skip-gram Baselines. We use PubMed corpus, which consists of 29 million biomedical abstracts, to train SG_W. For SG_S and $SG_{S.C}$, we further utilize the annotations provided in Pubtator (Wei et al., 2013). The annotations (names and their associated concepts) come with five categories: disease, chemical, gene, species, and mutation. We use annotations of the two popular classes: disease and chemical. In preprocessing, text is tokenized and lowercased. Words that appear less than 3 times are ignored. We use spaCy library for this parsing. In total, our vocabulary contains approximately 3 millions words, 700 thousand names, and 85 thousand CUIs. We use Gensim library to train all the skip-gram baselines. The embedding dimension is 200, and the context window size is 6. Negative sampling is used with the number of negatives set to 5.

Biomedical Named Encoder (BNE). We set the character embedding dimension to 50, and initialize their values randomly. We use 200 dimensions for the outputted name embeddings. The hidden states' dimensions for both character and word-level BiLSTM are 200. We use Adam optimizer with the learning rate of 0.001, and gradient clipping threshold set to 5.0. Training batch size is 64. Dropout with the rate of 0.5 is used to regularize the model. Average performance on validation sets of biomedical name normalization experiment (see Section 4.3) is used as a criteria to stop the model training.

Training of BNE. Our proposed model is trained using only the synonym sets in UMLS[2], *i.e.,* $\mathcal{U} = \{\mathcal{S}_c\}$. We limit the synonyms to those of disease concepts[3]. We intentionally leave the chemical concepts out for out-domain evaluation. As a result, approximately 16 thousand synonym sets (associated to that number of disease concepts) are collected for training. These synonym sets include 156 thousand disease names in total. In each training batch, one positive and one negative pairs are sampled separately for each loss. The pre-trained word (or name/concept) embeddings are taken from the skip-gram baselines as described before. We denote two configurations, associated to Options 1 and 2 (see Section 3.2), as **BNE + SG$_{\mathbf{S.C}}$** and **BNE + SG$_{\mathbf{W}}$**, respectively. Next, we present the evaluations of these models.

[2]We use the 2018AA version released in May, 2018.

[3]We consider the diseases that exist in the CTD's MEDIC disease vocabulary (Davis et al., 2014).

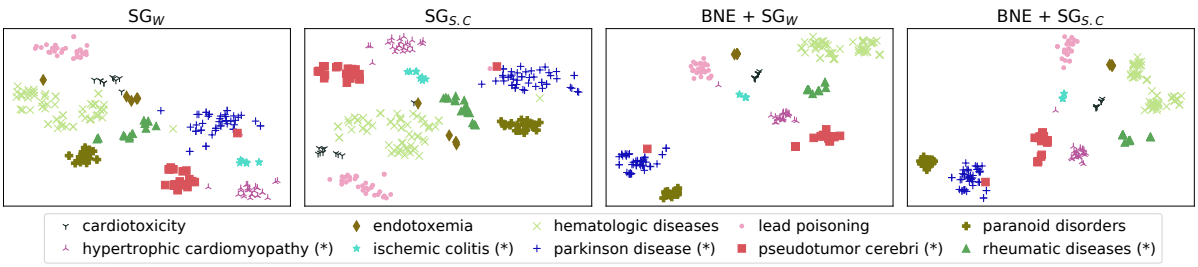
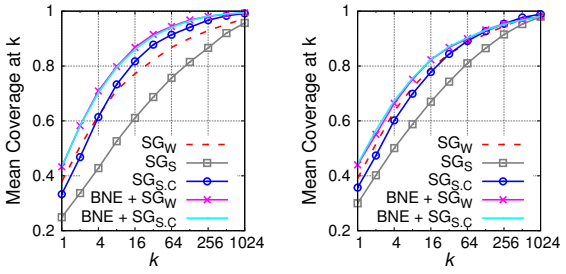


Figure 3: t-SNE visualization of 254 name embeddings. These names belong to 10 disease concepts in which 5 of these concepts appear in the training data, while the other 5 concepts (marked with (*)) do not. It can be observed that BNE projects names of the same concept close to each others. The model also retains closeness between names of related concepts, such as 'parkinson disease' and 'paranoid disorders' (see the blue and olive plus signs).



(a) Diseases (in-domain) (b) Chemicals (out-domain)

Figure 4: *Mean coverage at k*: average ratio of correct synonyms that are found in k-nearest neighbors, which are estimated by cosine similarity of name embeddings. Note that names in these disease and chemical test sets are not seen in the training data.

4.1 Closeness Analysis of Synonymous Embeddings

We propose a measure to estimate the closeness between name embeddings of the same concept. For each name, we consider its k most similar names estimated by cosine similarity of their embeddings. We define *coverage at k* as ratio of correct synonyms that are found in the k-nearest neighbors. We report the average score of all query names, as *mean coverage at k*.

We create two test sets for this experiment, one for disease names and one for chemical names. Given the CTD's MEDIC disease vocabulary, we randomly select 1000 concepts and all their corresponding names in UMLS. In this experiment, we exclude these 1000 concepts from the synonym sets used to train BNE encoder. Furthermore, to ensure the quality of the selected names, we only consider the ones that appear in the high-quality biomedical phrases collected by Kim et al. (2018). Similarly, we create another test set for chemical names. This chemical set is used to evaluate out-domain performance since our model is trained using only disease synonyms.

As shown in Figure 4, BNE outperforms other embedding baselines that do not consider the synonym-based objective. More importantly, the model also generalizes well to out-domain data (chemical names). Furthermore, among the skip-gram baselines, the context-based name embedding model (SG$_S$) is worse than the average word embedding baseline (SG$_W$). The result again indicates that words in biomedical names are more indicative about their conceptual identities.

The embedding plots in Figure 3 further illustrate the effectiveness of our encoder in enhancing the similarity between synonymous representations. By investigating name embeddings of an unseen concept 'pseudotumor cerebri', we observe that BNE is robust to the morphology of biomedical names, such as 'benign hypertension intracranial' and ' benign intracran hypt'. The model is also aware of word importance in long names such as 'intracranial pressure increased (benign)'. Moreover, since BNE is trained using synonym sets, the encoder is equipped with knowledge about alternative expressions of biomedical terms, *e.g.,* 'intracranial hypertension' and 'intracranial increased pressure'. The knowledge can be used to infer quality representations for new synonyms. However, similar to skip-gram baselines, BNE faces serious challenges if the names are unpopular and contain words that do not reflect their conceptual meanings. For example, for this 'pseudotumor cerebri' concept, the name "Nonne's syndrome"[4] is distant from its concept cluster (see the red square locating near the blue plus signs in Figure 3).

[4]Dr. Max Nonne coined the name 'pseudotumor cerebri' in 1904.

Models	NCBI (Disease)	BC5CDR (Disease)	BC5CDR (Chemical)
Jaccard	0.424	0.410	0.607
SG_W	0.499	0.494	0.598
SG_W + WMD	0.532	0.526	0.637
SG_S	0.487	0.472	0.623
$SG_{S.C}$	0.531	0.510	0.628
BNE + SG_W	<u>0.695</u>	<u>0.718</u>	<u>0.664</u>
BNE + $SG_{S.C}$	**0.713**	**0.734**	**0.672**

Table 2: Mean average precision (MAP) performance on the synonym retrieval task. The best and second best results are in boldface and underlined, respectively.

4.2 Synonym Retrieval

We evaluate the embeddings in synonym retrieval application: given a biomedical mention (or name), retrieving all its synonyms from a controlled vocabulary by ranking. We use NCBI-Disease (Doğan et al., 2014) and BC5CDR (Li et al., 2016) datasets in this evaluation. NCBI-Disease contains disease mentions extracted from PubMed abstracts, while BC5CDR contains both disease and chemical mentions. These mentions are used as queries in this synonym retrieval task. Note that, different from the closeness evaluation, a disease name may or may not appear in the synonym sets used to train BNE encoder. On the other hand, chemical queries are completely unseen during the model training. For each query, we retrieve a list of potentially associated concepts. A concept is retrieved if one of its names is similar to the query (estimated by BM25 score). We collect all names of the top-20 retrieved concepts as a synonym candidate set. Cosine similarity is then used to rank the candidates. We also evaluate the results with Jaccard and Word's Mover Distance (WMD) (Kusner et al., 2015) measures.

As shown in Table 2, SG_W+WMD outperforms Jaccard baseline (in MAP score), mainly because of its ability to capture semantic matching. However, both baselines are non-parametric. In contrast, BNE+SG_W learns additional knowledge about the synonym matching by using synonyms sets in UMLS as training data. Although the model is trained on only disease names, it also generalizes well to chemical names. Furthermore, comparing between the two configurations of BNE, both BNE+SG_W and BNE+SG_{SC} models yield comparable performances. However, BNE+SG_W is simpler since it does not require pre-trained name and concept embeddings.

Models	NCBI (Di)	BC5CDR (Di)	BC5CDR (Ch)
Jaccard	0.843	0.772	0.935
SG_W	0.800	0.725	0.771
SG_W + WMD	0.779	0.731	0.919
SG_S	0.815	0.790	0.929
$SG_{S.C}$	0.838	0.811	0.929
BNE + SG_W	0.854	0.829	0.930
BNE + $SG_{S.C}$	0.857	0.829	0.934
Wieting et al. (2015)	0.822	0.813	0.930
D'Souza and Ng (2015)	0.847	0.841	-
Leaman and Lu (2016)	<u>0.877</u>†	0.889†	0.941
Wright et al. (2019)	**0.878**†	0.880†	-
BNE + SG_W + XM	0.873	<u>0.905</u>	<u>0.954</u>
BNE + $SG_{S.C}$ + XM	<u>0.877</u>	**0.906**	**0.958**

Table 3: Name normalization accuracy on disease (Di) and chemical (Ch) datasets. The last row group includes the results of supervised models that utilize training annotations in each specific dataset. XM denotes the use of 'exact match' rule to assign the corresponding concept to a mention if the mention is found in the training data. † indicates the results reported by Wright et al. (2019).

4.3 Biomedical Name Normalization

Biomedical name normalization (a.k.a., biomedical concept linking) aims to map each biomedical mention appearing in text to its associated concept in a dictionary. We use NCBI-Disease and BC5CDR datasets in this evaluation. Similar to previous works, we use Ab3P (Sohn et al., 2008) to resolve local abbreviations. Composite mentions (such as 'pineal and retinal tumors') are split into separate mentions ('pineal tumors' and 'retinal tumors') using simple patterns as described in (D'Souza and Ng, 2015). For each mention, we find the concept CUI (in UMLS) that has the most similar name. The selected CUI is then mapped to its associated MeSH or OMIM ID in the CTD dictionary for evaluation. We only consider mentions whose associated concepts exist in the CTD dictionary and report the accuracy aggregated from all mentions in test set. Apart from existing baselines, we also re-implement compositional paraphrase model, proposed by Wieting et al. (2015). The difference is that we use word-level BiLSTM instead of recursive neural network. Furthermore, L_2 regularizations with the weights of 10^{-3} and 10^{-4} are applied on the BiLSTM's parameters and the difference between the trainable and initial word embeddings, respectively.

Different from the lexical (Jaccard) and semantic matching (WMD and SG_W) baselines, BNE ob-

tains high scores in both accuracy and ranking-based (MAP) metrics (see Tables 2, and 3). The result indicates that BNE has encoded both lexical and semantic information of names into their embeddings. Table 3 also includes performances of other state-of-the-art baselines in biomedical name normalization, such as sieve-based (D'Souza and Ng, 2015), supervised semantic indexing (Leaman and Lu, 2016), and coherence-based neural network Wright et al. (2019) approaches. Note that all these baselines require human annotated labels, and the models are specifically tuned for each dataset. On the other hand, BNE utilizes only the existing synonym sets in UMLS for training. When the dataset-specific annotations are utilized, even the simple exact matching rule can boost the performance of our model to surpass other baselines (see the last two rows in Table 3).

4.4 Semantic Similarity and Relatedness

We evaluate the correlation between embedding cosine similarity and human judgments, regarding semantic similarity and relatedness. Different from previous evaluations, this experiment aims to evaluate the conceptual similarity and relatedness. We use two biomedical datasets: MayoSRS (Pakhomov et al., 2011) and UMNSRS (Pakhomov et al., 2016). The former contains multi-word name pairs of related concepts, *e.g.,* 'morning stiffness' (C0457086) and 'rheumatoid arthriits' (C0003873). The latter contains only single-word name pairs and is spitted into similarity and relatedness partitions. For example, a pair with high similarity score are 'weakness' (C1883552) and 'paresis' (C0030552). For these two datasets, the names in each pair comes from different concepts, hence they do not appear in the synonym pairs used to train our encoder. Furthermore, the coverage of pre-trained word embeddings in baselines such as SG_W are 100% and 97% for UMNSRS and MayoSRS, respectively.

Table 4 shows that BNE models perform especially well on the multi-word relatedness test set (MayoSRS). Conceptual information has been utilized by these models to enrich the name representations. On the other hand, when the training is performed solely on the synonym pairs (only use $\mathcal{L}_{syn}$), the trained model is overfitted to the training task and do not generalize to other test cases.

SG_W is still a strong baseline in these benchmarks. Other skip-gram and fastText embeddings (Pakhomov et al., 2016; Chen et al., 2018), which are trained on a similar corpus, do not achieve better results. Beam et al. (2018) use a SVD-based word2vec model (Levy et al., 2015) to compute embeddings for biomedical concepts. Although the embeddings are trained on a much larger multimodal medical data, their results are lower than other baselines. Further investigation reveals that many concepts in the test sets do not exist in their pre-trained concept embeddings.

Models	UMNSRS (sim)	UMNSRS (rel)	MayoSRS (rel)
SG_W	0.645	0.584	0.518
Pakhomov et al. (2016)	0.620	0.580	-
Chen et al. (2018)	0.630	0.575	0.501
Beam et al. (2018)	0.411	0.334	0.427
SG_S	0.614	0.566	0.516
$SG_{S.C}$	**0.654**	0.592	0.557
BNE + SG_W	0.606	0.580	**0.626**
BNE + $SG_{S.C}$	0.637	**0.593**	0.602
BNE + $SG_{S.C}$ ($\mathcal{L}_{syn}$)	0.496	0.445	0.564
Wieting et al. (2015)	0.639	0.565	0.595

Table 4: Spearman's rank correlation coefficient between cosine similarly scores of name embeddings and human judgments, reported on semantic similarity (sim) and relatedness (rel) benchmarks.

5 Conclusion

By learning to encode names of the same concepts into similar representations, while preserving their conceptual and contextual meanings, our encoder is able to extract meaningful representations for unseen names. The core unit of our encoder (in this work) is BiLSTM. Alternatively, sequence encoding models such as GRU, CNN, transformer, or even encoders with contextualized word embeddings like BERT (Devlin et al., 2018), or ELMo (Peters et al., 2018) can be used to replace this BiLSTM, however, with additional computation cost. We also discuss different ways of representing the contextual and conceptual information in our framework. In implementation, we use the simple aggregation of pre-trained embeddings. The experiment results show that this approach is both efficient and effective.

Acknowledgments

We thank the anonymous reviewers for their insightful suggestions. This work was supported by Data Science & Artificial Intelligence Research Centre, NTU Singapore.

References

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. In *ICLR*.

Andrew L Beam, Benjamin Kompa, Inbar Fried, Nathan P Palmer, Xu Shi, Tianxi Cai, and Isaac S Kohane. 2018. Clinical concept embeddings learned from massive sources of medical data. *CoRR, abs/1804.01486*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *TACL*, 5:135–146.

Xiangrui Cai, Jinyang Gao, Kee Yuan Ngiam, Beng Chin Ooi, Ying Zhang, and Xiaojie Yuan. 2018. Medical concept embedding with time-aware attention. In *IJCAI*, pages 3984–3990.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *CoRR, abs/1803.11175*.

Qingyu Chen, Yifan Peng, and Zhiyong Lu. 2018. Biosentvec: creating sentence embeddings for biomedical texts. *CoRR, abs/1810.09302*.

Billy Chiu, Gamal Crichton, Anna Korhonen, and Sampo Pyysalo. 2016. How to train good word embeddings for biomedical nlp. In *BioNLP*, pages 166–174.

Kevin Clark, Minh-Thang Luong, Christopher D Manning, and Quoc Le. 2018. Semi-supervised sequence modeling with cross-view training. In *EMNLP*, pages 1914–1925.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *EMNLP*, pages 670–680.

Allan Peter Davis, Cynthia J Grondin, Kelley Lennon-Hopkins, Cynthia Saraceni-Richards, Daniela Sciaky, Benjamin L King, Thomas C Wiegers, and Carolyn J Mattingly. 2014. The comparative toxicogenomics database's 10th year anniversary: update 2015. *Nucleic acids research*, 43(D1):D914–D920.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *CoRR, abs/1810.04805*.

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 *ACL — IJCNLP*, volume 2, pages 297–302.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. 2015. Retrofitting word vectors to semantic lexicons. In *NAACL-HLT*, pages 1606–1615.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18(5-6):602–610.

Felix Hill, Kyunghyun Cho, and Anna Korhonen. 2016. Learning distributed representations of sentences from unlabelled data. In *NAACL-HLT*, pages 1367–1377.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *ACL — IJCNLP*, volume 1, pages 1681–1691.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In *ACL*, volume 1, pages 655–665.

Sun Kim, Lana Yeganova, Donald C Comeau, W John Wilbur, and Zhiyong Lu. 2018. Pubmed phrases, an open set of coherent phrases for searching biomedical literature. *Scientific data*, 5:180104.

Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Skip-thought vectors. In *NeurIPS*, pages 3294–3302.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *ICML*, pages 957–966.

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

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *TACL*, 3:211–225.

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):385.

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.

Miaofeng Liu, Jialong Han, Haisong Zhang, and Yan Song. 2018. Domain adaptation for disease phrase matching with adversarial networks. In *BioNLP*, pages 137–141.

Lajanugen Logeswaran and Honglak Lee. 2018. An efficient framework for learning sentence representations. *CoRR, abs/1803.02893*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NeurIPS*, pages 3111–3119.

Nikola Mrkšić, Ivan Vulić, Diarmuid Ó Séaghdha, Ira Leviant, Roi Reichart, Milica Gašić, Anna Korhonen, and Steve Young. 2017. Semantic specialization of distributional word vector spaces using monolingual and cross-lingual constraints. *TACL*, pages 309–324.

Serguei VS Pakhomov, Greg Finley, Reed McEwan, Yan Wang, and Genevieve B Melton. 2016. Corpus domain effects on distributional semantic modeling of medical terms. *Bioinformatics*, 32(23):3635–3644.

Serguei VS Pakhomov, Ted Pedersen, Bridget McInnes, Genevieve B Melton, Alexander Ruggieri, and Christopher G Chute. 2011. Towards a framework for developing semantic relatedness reference standards. *Journal of biomedical informatics*, 44(2):251–265.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL-HLT*, volume 1, pages 2227–2237.

Andreas Rücklé, Steffen Eger, Maxime Peyrard, and Iryna Gurevych. 2018. Concatenated p-mean word embeddings as universal cross-lingual sentence representations. *CoRR, abs/1803.01400*.

Richard Socher, Eric H Huang, Jeffrey Pennin, Christopher D Manning, and Andrew Y Ng. 2011. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *NeurIPS*, pages 801–809.

Sunghwan Sohn, Donald C Comeau, Won Kim, and W John Wilbur. 2008. Abbreviation definition identification based on automatic precision estimates. *BMC bioinformatics*, 9(1):402.

Sandeep Subramanian, Adam Trischler, Yoshua Bengio, and Christopher J Pal. 2018. Learning general purpose distributed sentence representations via large scale multi-task learning. In *ICLR*.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *ACL — IJCNLP*, volume 1, pages 1556–1566.

Ivan Vulić, Goran Glavaš, Nikola Mrkšić, and Anna Korhonen. 2018. Post-specialisation: Retrofitting vectors of words unseen in lexical resources. In *NAACL-HLT*, pages 516–527.

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.

Chih-Hsuan Wei, Hung-Yu Kao, and Zhiyong Lu. 2013. Pubtator: a web-based text mining tool for assisting biocuration. *Nucleic acids research*, 41(W1):W518–W522.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2015. From paraphrase database to compositional paraphrase model and back. *TACL*, 3:345–358.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2016. Towards universal paraphrastic sentence embeddings. In *ICLR*.

John Wieting and Kevin Gimpel. 2017. Revisiting recurrent networks for paraphrastic sentence embeddings. In *ACL*, pages 2078–2088.

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

‘