Journal of the American Medical Informatics Association, 26(5), 2019, 438–446
doi: 10.1093/jamia/ocy189
Advance Access Publication Date: 26 February 2019
Research and Applications

Research and Applications

deepBioWSD: effective deep neural word sense disambiguation of biomedical text data

Ahmad Pesaranghader,[1,2] Stan Matwin,[1,2] Marina Sokolova,[2,3,4] and Ali Pesaranghader[3]

[1]Faculty of Computer Science, Dalhousie University, Halifax, NS B3H 4R2, Canada, [2]Institute for Big Data Analytics, Dalhousie University, Halifax, NS B3H 4R2, Canada, [3]School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, ON K1N 6N5, Canada, and [4]School of Epidemiology and Public Health, University of Ottawa, University of Ottawa, Ottawa, ON K1G 5Z3, Canada

Corresponding Author: Ahmad Pesaranghader (ahmad.pgh@dal.ca)

Received 21 September 2018; Revised 3 December 2018; Editorial Decision 6 December 2018; Accepted 19 December 2018

ABSTRACT

Objective: In biomedicine, there is a wealth of information hidden in unstructured narratives such as research articles and clinical reports. To exploit these data properly, a word sense disambiguation (WSD) algorithm prevents downstream difficulties in the natural language processing applications pipeline. Supervised WSD algorithms largely outperform un- or semisupervised and knowledge-based methods; however, they train 1 separate classifier for each ambiguous term, necessitating a large number of expert-labeled training data, an unattainable goal in medical informatics. To alleviate this need, a single model that shares statistical strength across all instances and scales well with the vocabulary size is desirable.

Materials and Methods: Built on recent advances in deep learning, our deepBioWSD model leverages 1 single bidirectional long short-term memory network that makes sense prediction for any ambiguous term. In the model, first, the Unified Medical Language System sense embeddings will be computed using their text definitions; and then, after initializing the network with these embeddings, it will be trained on all (available) training data collectively. This method also considers a novel technique for automatic collection of training data from PubMed to (pre)train the network in an unsupervised manner.

Results: We use the MSH WSD dataset to compare WSD algorithms, with macro and micro accuracies employed as evaluation metrics. deepBioWSD outperforms existing models in biomedical text WSD by achieving the state-of-the-art performance of 96.82% for macro accuracy.

Conclusions: Apart from the disambiguation improvement and unsupervised training, deepBioWSD depends on considerably less number of expert-labeled data as it learns the target and the context terms jointly. These merit deepBioWSD to be conveniently deployable in real-time biomedical applications.

Key words: word sense disambiguation, biomedical text mining, deep neural networks, bidirectional long short-term memory network, zero-shot learning

INTRODUCTION

With recent advances in biomedicine, we see a massive amount of biomedical text data being generated every day. To gain knowledge from these data, developing natural language processing (NLP) tools that mine them accurately within a reasonable time is crucially important. NLP components that include named entity recognition programs,[1] syntactic parsers,[2] and relation extractors[3,4] build the foundation of many high-level biomedical information extraction and knowledge discovery applications.[5–8] Also, it is shown that the biomedical text data such as scientific articles,[9] clinical narratives,[10] and health-related social media posts,[11] abound with ambiguous terms (hereafter, instead of saying *ambiguous word* we use

ambiguous term because a [biomedical] conceptual unit that we try to disambiguate can be represented by a series of words; as in *malignant B-cell lymphoma* or *benign B-cell lymphoma* for the target ambiguous term *B-cell lymphoma*). In the lowest level, surrounded by this innate ambiguity, all other components and the full biomedical application will suffer if it is not resolved properly.

A word sense disambiguation (WSD) algorithm attempts to predict the correct sense of a term within a *context* given a set of candidates. For example, in the sentence "Ca intakes in the United States and Canada appear satisfactory among young adults," the sense set for *Ca* consists of *Canada* (s_1), *California* (s_2), *calcium* (s_3), and *cornu ammonis* (s_4) and the goal is to predict the correct sense s_3 for this specific occurrence of *Ca*. It is shown that this automatically identifying the intended sense of ambiguous words improves the performance of clinical and biomedical applications such as medical coding and indexing,[12,13] detection of adverse drug event,[14] automatic medical reporting,[15,16] and other secondary uses of data such as information retrieval and extraction,[17] and question-answering systems.[18] These capabilities are becoming essential tasks due to the growing amount of information available to researchers, the transition of healthcare documentation and patient-practitioner interaction toward electronic health records and automatic expert systems, and the push for quality and efficiency in health care.

Supervised machine learning WSD algorithms typically build 1 separate classifier for each ambiguous term, which will be trained solely on the instances of that term. That is, to train an accurate WSD model, a large amount of annotated instances are needed, the curation of which will be expensive and labor-intensive particularly in health informatics.[19,20] Recent studies in the biomedical domain incorporate expert-involved active learning techniques to accelerate the labeling process of this training data.[21,22] Nevertheless, considering the multiclassifier design of the traditional supervised WSD models, the real-world implementation of them in the domain is still impracticable.

We introduce a 1-size-fits-all deepBioWSD architecture for disambiguation of biomedical text data, a deep learning–based model that unifies all disambiguation classifiers into 1 single network. In a supervised manner, this network will be trained on all existing instances of the ambiguous terms as 1 group of training data in which sense-context pair and $s_i \epsilon \{1.0,\ 0.0\}$ constitute the input and the output, respectively. While the network encodes the shared information among all instances, for a given training-instance, it learns the senses of the unlabeled terms in the context and the sense of the labeled center term at the same time. To this end, our architecture employs a bidirectional long short-term memory network (BLSTM), and works with neural sense embeddings, which can be pretrained.

Supervised WSD in biomedicine

Jimeno-Yepes et al[23] prepared the National Library of Medicine's MSH WSD dataset in 2011 with naive Bayes accuracy of 93.84% (NB [these abbreviations are used during evaluation of the WSD algorithms]). Later, traditional discriminative models with rigorous linguistic and biomedical specific features were used for WSD evaluation.[24,25] To avoid an intense feature engineering, recently, the state-of-the-art accuracy of 95.97% was achieved by Jimeno-Yepes[26] using unigrams and word embeddings with support vector machines (SVM_{Yepes}); they also reported the accuracy of 94.87% for their long short-term memory networks (LSTMs). In another supervised model, Antunes and Matos[27] used bag-of-words as local features and word embeddings as global features and reported accuracy of 95.6% when SVM classifiers were employed ($SVM_{Ant\text{-}Mat}$). To eliminate an extreme need for extensive amount of annotated data to train classifier of each term, Sabbir et al[28] recently developed a knowledge-based model at the cost of accuracy (92.24%, KN). In another recent knowledge-based study, Duque et al[29] reported accuracy of 71.52% on MSH WSD for their system called Bio-Graph that employs a PageRank algorithm to work with occurrence graphs built from Medline abstract to address WSD (Bio-Graph).

Neural embeddings for WSD

With recent interests in training neural word embeddings from large raw corpora,[30–32] several studies included pretrained word embeddings in their WSD models, some of which were concerned with biomedical text.[33–36] Lately, computation of sense embeddings has gained the attention of researchers as well where their importance in the WSD tasks has been investigated;[37–40] however, the mapping of these hardly interpretable inducted sense embeddings to a sense inventory (eg, the Unified Medical Language System [UMLS]) has been the main bottleneck for their wider employment in WSD systems.[41] In the deepBioWSD model, first, we build our sense embeddings using the UMLS text definition of senses; then, these embeddings initialize our BLSTM network before training.

Bidirectional LSTM

LSTMs address the vanishing gradient problem in RNNs by incorporating gating functions into their state dynamics (see Supplementary Appendix).[42] Standard Recurrent Neural Networks (RNNs) and LSTMs, however, have restrictions as the future input information cannot be reached from the current state, so, a Bidirectional LSTM fuses 1 forward and 1 backward LSTM.[43] In WSD, this means we are able to encode the information of both preceding and succeeding words with respect to a pivotal ambiguous term. Kågebäck and Salomonsson[35] proposed a partially shared multiclassifier WSD model with BLSTMs that employed word embeddings ($BLSTM_{Kåg\text{-}Sal}$). In our previous work, we developed a single-classifier WSD model with just 1 BLSTM network ($BLSTM_{Pes\text{-}etal}$);[36] this model, however, uses 2 separate word and sense spaces for the context and center words, which caused inconsistency and worse performance. As we will see, the deepBioWSD network is only dependent on sense space for both center and context terms, an architectural improvement over $BLSTM_{Pes\text{-}etal}$ network for better sense prediction, faster training, and less dependency on expert-labeled data. Other existing BLSTM-based WSD algorithms are *Seq2Seq*-inspired models, which typically underperform conventional supervised WSD models.[44–46]

Zero-shot learning

Zero-shot learning (ZSL) aims at predicting labels for instances that belong to classes that were not directly seen during training.[47,48] The underlying secret ensuring the success of ZSL is to find an intermediate semantic representation to transfer the knowledge learned from seen classes to unseen ones.[49] The scalability of the model is of utmost importance since a large amount of unlabeled data is generally present and can be received by interaction with the environment,[50] which is the case in medical informatics. We show deepBioWSD with a unitary and uniform network architecture that it offers benefits from ZSL; that also, in turn, prevents the "cold start" problem (ie, when a model cannot draw any inferences as it has not yet gathered sufficient information related to a subject matter or application; hence, training of the model from scratch with

sufficient amount of labeled data seems inevitable) that exists in other supervised WSD algorithms.

Experimental data

Unified Medical Language System

The UMLS (https://www.nlm.nih.gov/research/umls/) is a terminology integration system that contains Metathesaurus and SPECIALIST Lexicon. The Metathesaurus holds ~3.4 million biomedical and clinical concepts (hereafter, we use *concept* and *sense* [of a term] interchangeably) by maintaining their hierarchical relationships. Each concept has a unique identifier called CUI (Concept Unique Identifier), a set of representative *terms*, and a text definition. The Metathesaurus provided us with the sense sets of the ambiguous terms. The SPECIALIST Lexicon resource contains information about common English vocabulary and biomedical terms by offering tools for language processing. We used its programs to demarcate terms in the contexts; in our early example, *the United States* is an unambiguous term (CUI: C0041703) consisting of 3 words, and *satisfactory* is a single-word ambiguous term (C0205410, C1547307). The latest UMLS release 2018AA was used in the study. This release covers >83 000 ambiguous representative terms.

Medline abstracts

Medline includes over 20 million citations of life sciences and biomedical articles from 1966 to the present. Combined with the UMLS concept definitions, we employed Medline 2013 bigram-list (https://mbr.nlm.nih.gov/Download/) to create our sense embeddings.

Validation datasets

We employed the MSH WSD dataset (https://wsd.nlm.nih.gov/collaboration.shtml) for the evaluation of WSD algorithms.[23] This dataset provides 37 888 instances for 203 ambiguous terms (including abbreviations) that take 2–5 senses (~100 instances per each sense are provided). Prepared from Medline, every instance of a target ambiguous term is manually annotated with a CUI within the sense set of that term. For example, an instance of *Ca* is labeled with either C0006823 (Canada), C0006675 (California), C0006754 (calcium), or C3887642 (cornu ammonis); while every instance of the target term *lymphogranulomatosis* takes the sense C0036202 (benign lymphogranulomatosis) or C0019829 (malignant lymphogranulomatosis).

MATERIALS AND METHODS

Pretraining of sense embeddings

Inspired by studies for (high-dimensional) distributed representation of biomedical concepts,[51–53] and low-dimensional vector representation of words,[54,55] we pretrained UMLS sense embeddings in 6 steps as depicted in Figure 1. In essence, the second-order computation of vector representation of concepts prevents the issue of sparsity (of word features) in the first-order vector representation of their definitions, pointwise mutual information statistically defines the degree of relevance between each biomedical concept and its (second-order) word features, and latent semantic analysis aims at condensing the final high-dimensional vectors to a size proper for a deep neural network. These steps briefly explained below are executed in advance to compute sense embeddings of the UMLS concepts before training our deepBioWSD network which they initialize (see Supplementary Appendix for further details).

Step 1—Bigrams and Medline words co-occurrence matrix.

We built a co-occurrence matrix from the bigram-list of Medline abstract. This matrix is symmetric and sparse, and represents the contextual information of the Medline words.

Step 2—UMLS concept definition extension and definition matrix. The definition extension of concepts by their immediate concepts' in an ontology/thesaurus enriches their semantic.[51,56] When applied to the UMLS concepts, words in the extended definitions have associated co-occurrence vectors from Medline computed in step 1. For every (extended) definition, the definition matrix stores the frequency of these word features.

Step 3—Second-order co-occurrence (SOC) matrix. To build a SOC vector of a concept, we first summed the Medline co-occurrence vectors of the content words in that concept's extended definition, and then normalized the result vector by the number of words in the definition. In other words, we took the centroid of the vectors associated with each word in the definition, and then normalized the result to uniformly treat the different size definitions.

Step 4—Pointwise mutual information (PMI) on SOC matrix. Not all word features associated with a concept are equally important. PMI, as in equation 1, statistically measures the level of association between the concepts (their associated words; ie, $word_i$) and the word features (ie, $word_j$), instead of naive consideration of word feature frequency cutoff threshold.[57,58] Once PMI values are calculated—with respect to the (frequency) probabilities of (co-)occurrences of these words, our validation set helps to set a low cutoff threshold for the removal of irrelevant features. We applied the add-1 smoothing technique to the SOC matrix in advance to avoid bias toward infrequent occurrences.[52]

$$PMI_{sense}(word_i, word_j) = log\frac{p(word_i, word_j)}{p(word_i) \times p(word_j)} \quad (1)$$

Step 5—Latent semantic analysis (LSA) on PMI-on-SOC matrix—LSA, given by equation 2, uses a singular value decomposition algorithm that resulted 2 square and unitary matrices $\boldsymbol{U}$ an $\boldsymbol{V}^T$, and a non-negative diagonal matrix $\boldsymbol{\Sigma}$ that held singular values on its diagonal in a nonincreasing order.[59]

$$PMI\_on\_SOC = \boldsymbol{\Sigma V}^{\mathrm{T}} \quad (2)$$

Step 6—Reducing the rank of singular values—Having equation 3, we truncated the singular value decomposition to 100 for low-dimensional representation of UMLS concepts. Determined by our validation set, smaller embedding sizes yielded worse WSD results, and higher dimensions did not improve the accuracy and just increased the training time.

$$sense\_embeddings = \boldsymbol{U\Sigma}_{100} \quad (3)$$

deepBioWSD network definition

In contrast to other supervised WSD networks, in which a softmax layer with a cross-entropy or hinge loss is often parametrized to select the corresponding weight matrix and bias vector for every sense of an ambiguous term, our network shares parameters over all senses. Given an instance and the position of a target term, the deepBioWSD network computes a probability distribution over candidate senses of that term.

The architecture of our network consists of 7 layers (Figure 2). Due to the replacement of the conventional softmax layer with a linear (regression) layer, we imposed a modification to the input. That is, apart from the contextual features, the sense for which we want

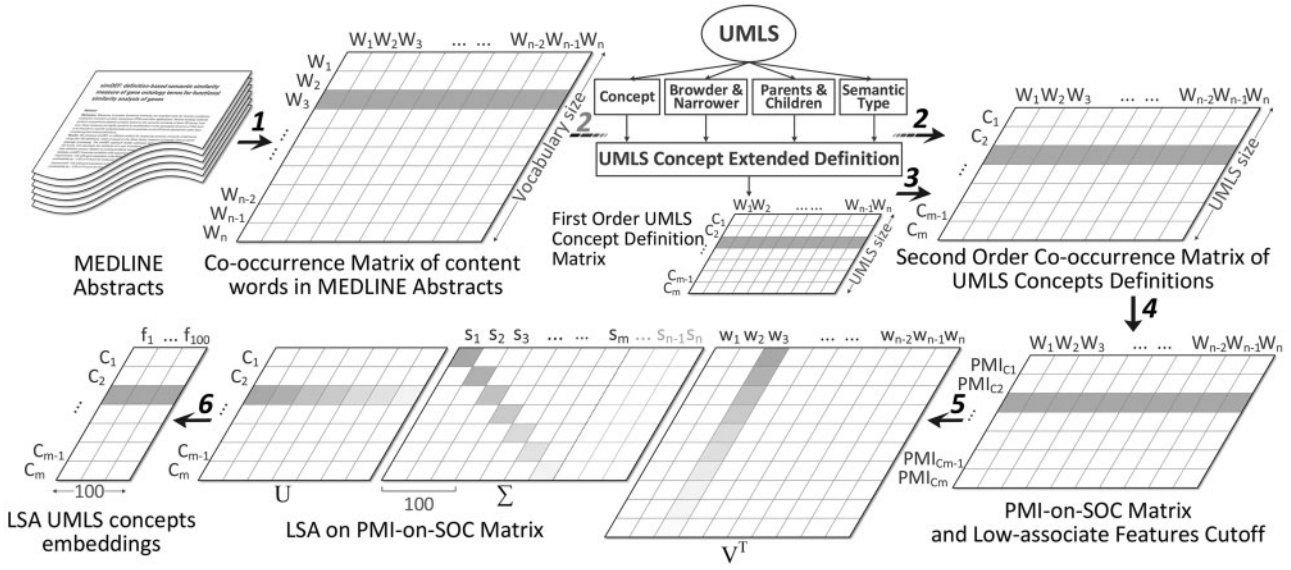

Figure 1. The figure represents different steps in our unsupervised method to generate low-dimensional sense embeddings for the Unified Medical Language System (UMLS) concepts. These embeddings initialize of disambiguation deep neural network. C: concept; f: new feature; LSA: latent semantic analysis; PMI: pointwise mutual information; S: salient feature; SOC: second-order co-occurrence; W: word feature.

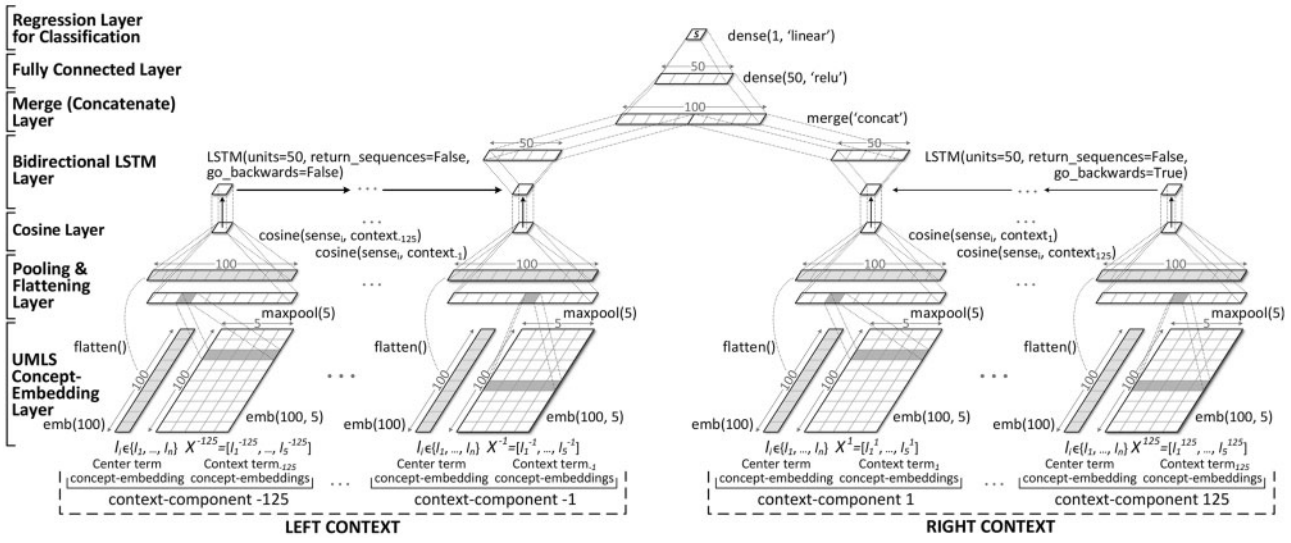

Figure 2. The figure illustrates our 1-size-fits-all deepBioWSD network which treats all center ambiguous terms (and their instances) uniformly. The *emb* represents *embedding size*, l_i is the *current candidate sense* (or *label*) under investigation, and X^j is the jth term in the context (left or right). Besides training on the center terms, the embeddings of the context terms would be updated and learned (ie, disambiguated) during training. LSTM: long short-term memory network; UMLS: Unified Medical Language System.

to discover whether the given context is meaningful will be provided as input. For an ambiguous term with the sense set $\{s_1, \ldots, s_n\}$, the network runs n times (for every sense) and the highest-confidence sense would be selected. In lower layers, to determine proximity of the senses and the given context, after computing cosine similarities of each candidate sense (embedding) with the senses of the context terms, the sequential result of the cosine similarities between the correct sense and the surrounding context communicate a pattern-like information that our BLSTM layer encodes—which consequently yields higher confidence in the upper regression layer; however, for the incorrect senses, this premise of homogeneity and proximity does not hold (ie, negative samples). Several studies already incorporated the idea of sense-context cosine similarities in their WSD models.[36,60–62] Nevertheless, the context terms, which are determined by the SPECIALIST Lexicon during the disambiguation process, can be ambiguous themselves. To deal with their ambiguity, just before the cosine layer, a pooling layer is devised, the result of which learns the senses of the ambiguous terms appeared in the context. This means the network takes gradients with respect to (shared) sense embeddings of both the target term and the context terms at the same time.

UMLS Concept Embedding Layer. For one instance, the input of the network consists of a sense and a list of (left and right) context terms, which paired together form a list of context components. For context D, which encompasses an ambiguous term with the sense set of $\{s_1, \ldots, s_n\}$, the embedding layer weights for the examined

input sense s_i, is determined by equation 4. Then, this input is copied to $|D|$ positions of the context to form the first pair of the context-components and set the same embedding weights in the layer.

$$\boldsymbol{l}_i = \boldsymbol{W}_s^l.\boldsymbol{v}_s(\boldsymbol{s}_i),\ i \in \{1,\ \ldots,\ \mathrm{n}\} \tag{4}$$

where, $\boldsymbol{l}_i \in \mathbb{R}^{100}$ and $\boldsymbol{v}_s(\boldsymbol{s}_i)$ is the 1-hot representation of the sense. A 1-hot representation is a vector with the dimension $\boldsymbol{V}_s$ consisting of $|\boldsymbol{V}_s| - 1$ zeros and a single one that indicates the index of a sense in a look-up table; the $\boldsymbol{V}_s$ size is equal to the number of CUIs in the UMLS. The $\boldsymbol{W}_s^l$ is the shared look-up table for the center terms and context terms; it is initialized with the sense embeddings that we computed in advance. Equation 4 have the effect of picking the column (ie, a sense embedding) from $\boldsymbol{W}_s^l$ corresponding to that sense.

Regarding a context term input, which form the second pair of a context component, at position k in the same context D the embedding weights are determined by

$$\begin{aligned} \boldsymbol{x}^k &= [\boldsymbol{l}_1,\ \ldots,\ \boldsymbol{l}_n] \in \mathbb{R}^{100\times m}, \\ k &\in \left\{-\frac{|D|}{2},\ \ldots, -2, -1,\ 1,\ 2,\ \ldots,\ \frac{|D|}{2}\right\} \end{aligned} \tag{5}$$

where, $\boldsymbol{l}_i$ is set by equation 4, and k is the position of the term in the context (left or right) while $|D|/2 = 125$ is a hyperparameter of the network (padding or truncating was applied wherever needed). m is another hyperparameter that typically should be equal to the size of the largest sense set; however, in the experiments of the study we observed an inverse relationship between the sense set size and the occurrence frequency of the terms, therefore we limited m to be 5. This means only those terms in the context were inputted to the network that had the sense set of size 5 or less (ie, some infrequent terms were ignored). This resulted a faster convergence with no accuracy loss. For those terms with the sense set size of <5, a generic embedding vector of very large negative numbers was employed to fill in the void senses; this helped *maxpooling* consider only the sense embeddings of a context term.

Pooling and Flattening Layer. Here, max operation is applied over all rows per each context term's sense embeddings, denoted as maxpool(5) in Figure 2. After maxpooling, each context term is represented with a 100-dimensional global feature vector. We also flattened the result column vector into a row vector as an integrated part of the maxpooling layer; that is, at position k in the context, the pooling and flattening layer gives $\overline{\boldsymbol{l}}^k \in \mathbb{R}^{1\times 100}$ for a target term sense and $\overline{\boldsymbol{x}}^k \in \mathbb{R}^{1\times 100}$ for the predicted context term sense. Despite the intuitive use case of maxpooling to deduce the proper sense of a context term, experimentally it worked better than *averagepooling*.

Cosine Layer. In $|D|$ positions of context components, the cosine similarities between the embedding vector of the examined sense and the maxpooled of the context terms are calculated. Computed by equation 6, the results are 2 row-vectors of size $|D|/2$ each containing the cosine similarities of the context components of their side:

$$\boldsymbol{c}_{\boldsymbol{left}} = [\boldsymbol{c}_l^k,\ \ldots,\ \boldsymbol{c}_l^{-1}] \in\ \mathbb{R}^{1\times|D|/2},\ k \in \left\{-\frac{|D|}{2},\ \ldots, -2, -1\right\} \tag{6}$$

$$\boldsymbol{c}_{\boldsymbol{right}} = [\boldsymbol{c}_r^1,\ \ldots,\ \boldsymbol{c}_r^k] \in\ \mathbb{R}^{1\times|D|/2},\ k \in \left\{1,\ 2,\ \ldots,\ \frac{|D|}{2}\right\}$$

$$c^i = cosine(\overline{\boldsymbol{l}}^i, \overline{\boldsymbol{x}}^i) = \frac{\sum \overline{\boldsymbol{l}}^i \odot \overline{\boldsymbol{x}}^i}{\|\overline{\boldsymbol{l}}^i\| \times \|\overline{\boldsymbol{x}}^i\|},\ k \in \left\{1,\ \ldots,\ k\right\}$$

Bidirectional LSTM Layer. With 1 forward and 1 backward LSTM networks, we have a left context–dedicated LSTM network that receives the cosine similarities from left to right, and right context–dedicated LSTM network that receives the cosine similarities from right to left. $\boldsymbol{c}_{\boldsymbol{left}}$ and $\boldsymbol{c}_{\boldsymbol{right}}$ are the inputs of these networks; their outputs are the vectors $\boldsymbol{h}_{\boldsymbol{left}} \in \mathbb{R}^{1\times 50}$ and $\boldsymbol{h}_{\boldsymbol{right}} \in \mathbb{R}^{1\times 50}$, each encoding the received information from one side of the target ambiguous term (50 is another hyperparameter).

Concatenation Layer. This layer concatenates the output row vectors of the BLSTM layer:

$$\boldsymbol{h}_{\boldsymbol{merge}} = [\boldsymbol{h}_{\boldsymbol{left}}, \boldsymbol{h}_{\boldsymbol{right}}] \in \mathbb{R}^{1\times 100} \tag{7}$$

Fully Connected Layer. Further, for a better representation, a hidden fully connected layer $\boldsymbol{h}_{\boldsymbol{fc}}$ is devised which is:

$$\boldsymbol{h}_{\boldsymbol{fc}} = ReLU(\boldsymbol{h}_{\boldsymbol{merge}}\ .\ \boldsymbol{W}_h + \boldsymbol{b}_h) \in \mathbb{R}^{1\times 50} \tag{8}$$

where, *ReLU* is rectified linear unit function;[63] $\boldsymbol{W}_h \in \mathbb{R}^{100\times 50}$ and $\boldsymbol{b}_h \in \mathbb{R}^{1\times 50}$ are the weights and bias for the hidden layer. The result of this layer embeds the input sequence into a vector of size 50.

Regression for Classification Layer. This layer outputs a single value that is computed by:

$$\hat{y}_{s_i} = \boldsymbol{h}_{\boldsymbol{fc}}.\boldsymbol{W}_{\boldsymbol{out}}\ + \boldsymbol{b}_{\boldsymbol{out}},\ s_i \in \{s_1,\ \ldots,\ s_n\} \tag{9}$$

where, $\boldsymbol{h}_{\boldsymbol{fc}}$ comes from the previous layer, and $\boldsymbol{W}_{\boldsymbol{out}}$ and $\boldsymbol{b}_{\boldsymbol{out}}$ are the weights and bias of this *linear* layer.

During network training, for an instance with its given context and the correct sense as inputs, $\hat{y}_{s_i}$ is set to be **1.0**, whereas for the same context with the incorrect senses it is set to be **0.0**. During testing, however, among all the senses, the output of the network for a sense that gives the highest value of $\hat{y}_{s_i}$ is considered as the true sense of the ambiguous term. In other words, the correct sense is:

$$\underset{s_i}{argmax}\{\hat{y}_{s_1}, \ldots, \hat{y}_{s_n}\},\ s_i \in \{s_1,\ \ldots,\ s_n\} \tag{10}$$

By applying softmax to the results of the estimated values $\{\hat{y}_{s_1}, \ldots, \hat{y}_{s_n}\}$, we can represent them as probabilities. This facilitates interpretation of them especially when deepBioWSD is benefiting from an active learning setting where intricacy and importance of 1 instance are measured.

The final recommended hyperparameters of the network which were determined during the validations are provided in Supplementary Appendix.

Unsupervised collection of training data

Considering the uniform structure of deepBioWSD, we also aimed at collecting more training data on which deepBioWSD could be pretrained. For this purpose, we employed Entrez Direct (https://www.ncbi.nlm.nih.gov/books/NBK179288/) to automatically gather data from PubMed. So, we devised a *query management* scheme that benefited from the notion of polyonymy of a concept (polyonomy is the employment of multiple names for the same concept): besides ambiguous representative terms, usually, one concept has other representative terms that are unambiguous (eg, *lymphogranulomatosis* vs *malignant lymphogranulomatosis*). By sending queries to PubMed for these unambiguous terms, we obtain abstracts for which we already know the true sense. It allowed us to create samples of unsupervised instances in a large quantity (see Supplementary Appendix). For each (unambiguous) sense query, we only considered the first 500 instances retrieved from PubMed (excluding the MSH WSD instances). A total of 180 175 instances were automatically prepared as PubMed returned <500 abstracts for some sense queries.

RESULTS

Sense similarity of pretrained embeddings

We will represent our method for pretraining of sense embeddings (see subsection Pre-training of sense-embeddings) plays an important role in sense prediction. This method organized the concepts based on their semantics (a physical library would be an appropriate analogy for this attempt), which will be later on introduced to the deepBioWSD network for a better and faster training. Table 1 represents a (cosine) sense similarity example for the ambiguous term *CP* (computed over the pretrained sense embeddings; ie, books in the library). Each column header represents one sense of *CP*, and the listed terms below are the closest UMLS concepts to that meaning of *CP*. In the table, instead of unfamiliar sense CUIs, the selected representative terms of the concepts are shown. Providing just 1 example here, we observed other senses followed the same sense similarity organization in the sense embedding space as well (see Supplementary Appendix for more examples).

Table 1. Sense similarity for candidate senses of the ambiguous term *CP*

Cerebral Palsy	*Propionibacterium acnes*	Cleft Palate
Convulsion	*Staphylococcus*	Glossoptosis
Spastic syndrome	*Propionibacterium*	Cleft Lip
Muscle Dystonia	*Stomatococcus*	Omodysplasia
Dysdiadochokinesis	*Micrococcus*	Congenital Megacolon
Choreoathetosis	*Flavobacterium*	Ectromelia
Quadriplegia	*Neisseriaceae*	Polydactylism
Trismus	*Acidovorax*	Teething
Hemiplegia	*Abiotrophia*	Congenital Aniridia
Muscle Hypertonia	*Paenibacillaceae*	Omphalocele
Muscle Spasticity	*Helicobacter*	Syndactyly

First WSD experiment: direct learning from center terms

Between-all-models comparisons: Table 2 compares the deepBioWSD with the other WSD algorithms. Despite those for which we already had the accuracy results on MSH WSD dataset, $BLSTM_{Kåg\text{-}Sal}$ and $BLSTM_{Pes\text{-}etal}$ were reimplemented with their best hyperparameters chosen, a few of which were slightly different from their original papers (eg, different context size). What we report here for deepBioWSD is based on 10-fold validation experiments we conducted after considering training, validation, and test splits; other models might not necessarily follow this strategy.

Supervised. Instances of every 203 term in MSH WSD data were divided into 10 nonoverlapping folds in which 1 fold was put aside for a final testing in a 10-time validation. Training on the rest of the 9 folds, we first randomly selected 5% as a validation set to tune hyperparameters and to find the proper number of epoch the network needed to train. After hyperparameters were chosen, the final model was trained on the whole training set (including the validation set), and then was evaluated on the 203 test data folds taken out already. In the experiments, macro and micro accuracies were considered for hyperparameter tuning as well as for the final evaluation of the test data (see Supplementary Appendix). After computing the test results of the all 10 times of validation, their average was considered as the results of the models.

Unsupervised. After finding the proper structure of the network, we experimented with 2 scenarios. First, the network was trained on the automatically collected data where the MSH WSD instances made the test data. Second, the network was pretrained on these unsupervised data and then it was retrained and evaluated according to the supervised layout described previously.

These results indicate the importance of pretrained sense embeddings initializing the network. Their influence, however, is inconsiderable when the network is pretrained on the unsupervised training data. In that case, the network produces sense embeddings from scratch, and the final updated embeddings are the byproduct of the network. Overall, deepBioWSD's single network architecture outperforms unsupervised KB and (multiclassifier) supervised WSD algorithms in the biomedical WSD task. Regarding training time, deepBioWSD also showed better efficiency (see Supplementary Appendix).

Within-our-model comparisons: We also studied if the flow of cosine similarities between a true sense and its preceding and succeeding terms (their senses) carried a sequential information that 1 BLSTM could encode and learn from. Therefore, according to what Table 3 represents, we introduced some changes in the input or in the structure of the network to verify that. We observed if we reverse the sequential flow of information into our BLSTM, we shuffle the order of the context terms, or replace our LSTMs with 2 fully connected networks of the same size 50, the achieved results were notably less than our original structure. Expectedly, due to a variable size of the original contexts (which forced padding/truncation), replacement of LSTMs with BLSTMs had negative effects.

Table 2. Accuracy results for MSH WSD dataset

Method	Algorithm	Macro Accuracy (%)	Micro Accuracy (%)
Unsupervised	Bio-Graph	71.52	–
	KB	92.24	–
	deepBio $WSD_{with\ random\ embeddings}$	92.16	91.93
	deepBio $WSD_{with\ pretrained\ embeddings}$	92.67[a]	92.51[a]
Supervised	NB	93.84	–
	$SVM_{Ant\text{-}Mat}$	95.60	–
	LSTM	94.87	94.78
	SVM_{Yepes}	95.97	95.81
	$BLSTM_{Kåg\text{-}Sal}$	95.64	95.47
	$BLSTM_{Pes\text{-}etal}$	95.53	95.39
	deepBio $WSD_{with\ random\ embeddings}$	93.88	93.71
	deepBio $WSD_{with\ pretrained\ embeddings}$	96.14	95.96
	deepBio $WSD_{pretrained\ unsupervised\ w/o\ sense\ embdgs}$	96.64	96.47
	deepBio $WSD_{pretrained\ unsupervised\ w/\ sense\ embdgs}$	96.82[a]	96.64[a]

[a]We observe deepBioWSD outperforms other measures in both supervised and unsupervised WSD settings.

Second WSD experiment: indirect learning from context terms

Considering ZSL, we also experimented if training on 1 target term's instances led to indirect insights into other terms. As an example, assume we are training the *ventricles* instance, "Coronal measurements of both ventricles were similar when obtained by US and MRI

Table 3. deepBioWSD with other architectural settings

Algorithm	Macro Accuracy (%)
Full network in Figure 2	96.82[a]
BLSTM with reverse directions in Figure 2	93.86
BLSTM with a shuffled context	91.98
Fully connected layers instead of BLSTM layer	95.23
BLSTM on the left and BLSTM on the right	95.81

[a]We observe the presence and the correct direction of the BLSTM have an impact on the WSD accuracy.

Table 4. Accuracy results for indirect learning from the context terms

Stage	Supervised Setting	Macro Accuracy (%)	Micro Accuracy (%)
Before Training	deepBioWSD$_{\textit{with random embeddings}}$	49.37	49.53
	deepBioWSD$_{\textit{with pretrained embeddings}}$	65.46[a]	65.73[a]
After Training	deepBioWSD$_{\textit{with random embeddings}}$	67.32	66.92
	deepBioWSD$_{\textit{with pretrained embeddings}}$	82.08[a]	81.85[a]

[a]We observe when deepBioWSD is initialized with pretrained embeddings it performs significantly better than the random initialization of the embeddings for the same task of WSD.

images"; having *ventricles* (meaning cerebral ventricles here) as the target ambiguous term, we gain knowledge about the context terms as well, including *US* and *MRI*. In a new *US* instance, this insight helps the network to predict if *US* means *United States* or *ultrasonography*.

To investigate indirect learning, we randomly divided 203 number of MSH WSD terms into 10 nonoverlapping folds, and then held (instances of) 1 of the folds for testing (as unseen data) and the rest for training (10-time validation). We selected 5% of the training set as a validation set to tune hyperparameters. The final network was trained on the whole training set and then was evaluated on the test set (averaged the individual test results on the unseen target terms). Table 4 represents the average of the 10 times of validation. These results clearly represent the influence of pretrained sense embeddings on the predictions. More importantly, we observe, when deepBioWSD is not directly trained on 1 term's instances, the preserved statistical information learned from the context (and its maxpooled embeddings) guides the network for more accurate sense prediction of that term when located at the center. Furthermore, with the current state of the network, the model will not suffer from the cold start problem because the model has been gaining the momentum, and with less number of training data needed, it will be fully trained on unseen terms in short order as well.

Except for BLSTM$_{\text{Pes-etal}}$, for which the results of this experiment were completely random (in all cases), we could not envision and conduct the experiment for the other supervised algorithms due to their multiclassifier design.

DISCUSSION

The deepBioWSD introduces an unorthodox WSD network in which all conceptual pieces of the biomedical domain (ie, pivotal and contextual terms) are designed to be interconnected—pieces that constantly communicate information to solve the jigsaw puzzle of WSD. The network, however, found 2 types of instances challenging. First, when the syntactic structure—with similar semantic theme—surrounding the candidate senses were very similar (eg, *veterinary assistant* and *veterinary medicine* for the ambiguous term *veterinary*). Second, when the senses are semantically so close that they share the same immediate parent in the UMLS, or 1 term directly subsumes the other sense (as in senses for *borrelia*, *heregulin*, and *HGF*) (see Supplementary Appendix).

We let MeSH and SNOMED CT demarcate the context terms (following the previous studies).[60,64] We found however by adding more vocabularies from the UMLS, fewer context terms will be ignored during prediction as the model will be inclusive of more biomedical terms or senses. For example, the term *12-step program* appeared frequently in the context of *AA* when it meant *Alcoholics Anonymous* (another meaning is *amino acid*); however, *12-step program* belongs to neither MeSH nor SNOMED CT, whereas the National Cancer Institute ontology (NCI) covers it. This consideration of more vocabularies was helpful, as it slightly improved the results with a smaller context size needed. Nonetheless, with more vocabularies, the possible number of senses one term can take grows, which to some extent offsets the advantage of a smaller context size.

CONCLUSIONS

One future work direction can be consideration of other unsupervised biomedical sense embedding methods in the model. Adding an attention mechanism to the network architecture might further improve the disambiguation results as well.[65] Also, more comprehensive and systematic study for the collection of unsupervised training data is needed. The model can also be evaluated on an extrinsic task with real-world applications (eg, Clinical Information Extraction).[17]

FUNDING

The study has been funded in part by NSERC CREATE grant and the grant from Poland's National Scientific Center available to SM, and in part by NSERC Discovery Grants available to SM and MS.

AUTHOR CONTRIBUTORS

AhP conceived of the presented idea, developed the theory, and investigated the network architecture. AhP and AlP performed the computations and implementation. SM and MS provided guidelines as to the test data, and verified the analytical methods. SM encouraged AhP to investigate unsupervised training of the model. SM and MS supervised the findings of this work. All authors discussed the results and contributed to the final manuscript.

SUPPLEMENTARY MATERIAL

Supplementary material is available at *Journal of the American Medical Informatics Association* online.

Conflict of interest statement. The authors confirm that there is no competing interests.

REFERENCES

1. Habibi M, Weber L, Neves M, *et al*. Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics* 2017; 33 (14): i37–48. doi: 10.1093/bioinformatics/btx228

2. Garg S, Galstyan A, Hermjakob U, *et al*. Extracting biomolecular interactions using semantic parsing of biomedical text. In: *AAAI'16 Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. Palo Alto, CA: AAAI Press; 2016: 2718–2726; Phoenix, Arizona USA.
3. Lee K, Lee S, Park S, *et al*. BRONCO: Biomedical entity Relation ONcology COrpus for extracting gene-variant-disease-drug relations. *Database (Oxford)* 2016; 2016: 13.
4. Luo Y, Uzuner Ö, Szolovits P. Bridging semantics and syntax with graph algorithms—state-of-the-art of extracting biomedical relations. *Brief Bioinform* 2017; 18 (1): 160–78. doi: 10.1093/bib/bbw001
5. Laranjo L, Dunn AG, Tong HL, *et al*. Conversational agents in healthcare: a systematic review. *J Am Med Inform Assoc* 2018; 25 (9): 1248–58.
6. P, Tafti A, Badger J, LaRose E, *et al*. Adverse drug event discovery using biomedical literature: a big data neural network adventure. *JMIR Med Inform* 2017; 5 (4): e51.
7. Xie F, Lee J, Munoz-Plaza CE, *et al*. Application of text information extraction system for real-time cancer case identification in an integrated healthcare organization. *J Pathol Inform* 2017; 8 (1): 48. doi: 10.4103/jpi.jpi_55_17
8. Lee K, Shin W, Kim B, *et al*. HiPub: translating PubMed and PMC texts to networks for knowledge discovery. *Bioinformatics* 2016; 32 (18): 2886–8. doi: 10.1093/bioinformatics/btw511
9. Cameron D, Kavuluru R, Rindflesch TC, *et al*. Context-driven automatic subgraph creation for literature-based discovery. *J Biomed Inform* 2015; 54: 141–57. doi: 10.1016/j.jbi.2015.01.014
10. Kavuluru R, Rios A, Lu Y. An empirical evaluation of supervised learning approaches in assigning diagnosis codes to electronic medical records. *Artif Intell Med* 2015; 65 (2): 155–66. doi: 10.1016/j.artmed.2015.04.007
11. Sadah SA, Shahbazi M, Wiley MT, *et al*. Demographic-based content analysis of web-based health-related social media. *J Med Internet Res* 2016; 18: e148. doi: 10.2196/jmir.5327
12. Preiss J, Stevenson M. The effect of word sense disambiguation accuracy on literature based discovery. *BMC Med Inform Decis Mak* 2016; 16: 57. doi: 10.1186/s12911–016–0296–1
13. Mishra R, Bian J, Fiszman M, *et al*. Text summarization in the biomedical domain: a systematic review of recent research. *J Biomed Inform* 2014; 52: 457–67. doi: 10.1016/j.jbi.2014.06.009
14. Harpaz R, Callahan A, Tamang S, *et al*. Text mining for adverse drug events: the promise, challenges, and state of the art. *Drug Saf* 2014; 37 (10): 777–90. doi: 10.1007/s40264–014–0218-z
15. Cohen KB, Demner-Fushman D. *Biomedical Natural Language Processing*. London: John Benjamins; 2014.
16. Wang X, Peng Y, Lu L, *et al*. Tienet: text-image embedding network for common thorax disease classification and reporting in chest x-rays. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Piscataway, NJ: IEEE; 2018: 9049–58; Salt Lake City, Utah, United States.
17. Névéol A, Robert A, Grippo F, *et al*. CLEF eHealth 2018 Multilingual Information Extraction task Overview: ICD10 coding of death certificates in French, Hungarian and Italian. In: *CLEF 2018 Evaluation Labs and Workshop: Online Working Notes, CEUR-WS*; 2018; Avignon, France. Accessed October 2018.
18. Roberts K, Kilicoglu H, Fiszman M, *et al*. Automatically classifying question types for consumer health questions. *AMIA Annu Symp Proc* 2014; 2014: 1018–27.
19. Pilehvar MT, Navigli R. A large-scale pseudoword-based evaluation framework for state-of-the-art word sense disambiguation. *Comput Linguist* 2014; 40 (4): 837–81.
20. Yuan D, Richardson J, Doherty R, *et al*. Semi-supervised word sense disambiguation with neural models. *ArXiv160307012*; 2016:1374–85.
21. Wang Y, Zheng K, Xu H, *et al*. Interactive medical word sense disambiguation through informed learning. *J Am Med Inform Assoc* 2018; 25 (7): 800–8. doi: 10.1093/jamia/ocy013
22. Wang Y, Zheng K, Xu H, *et al*. Clinical word sense disambiguation with interactive search and classification. *AMIA Annu Symp Proc* 2016; 2016: 2062–71.
23. Jimeno-Yepes AJ, McInnes BT, Aronson AR. Exploiting MeSH indexing in Medline to generate a data set for word sense disambiguation. *BMC Bioinformatics* 2011; 12 (1): 223. doi: 10.1186/1471–2105–12–223
24. Determining the difficulty of Word Sense Disambiguation - ScienceDirect. Accessed March 5, 2018.
25. Berster B-T, Goodwin JC, Cohen T. Hyperdimensional computing approach to word sense disambiguation. *AMIA Annu Symp Proc* 2012; 2012: 1129–38.
26. Jimeno Yepes A. Word embeddings and recurrent neural networks based on Long-Short Term Memory nodes in supervised biomedical word sense disambiguation. *J Biomed Inform* 2017; 73: 137–47. doi: 10.1016/j.jbi.2017.08.001
27. Antunes R, Matos S. Supervised learning and knowledge-based approaches applied to biomedical word sense disambiguation. *J Integr Bioinform* 2017; 14 (4).
28. Sabbir A, Jimeno-Yepes A, Kavuluru R. Knowledge-based biomedical word sense disambiguation with neural concept embeddings. *Proc IEEE Int Symp Bioinforma Bioeng* 2017; 2017: 163–70.
29. Duque A, Stevenson M, Martinez-Romo J, *et al*. Co-occurrence graphs for word sense disambiguation in the biomedical domain. *Artif Intell Med* 2018; 87: 9–19.
30. Collobert R, Weston J. A unified architecture for natural language processing: deep neural networks with multitask learning. In: *Proceedings of the 25th International Conference on Machine Learning*. New York, NY: ACM; 2008:160–7.
31. Mikolov T, Chen K, Corrado G, *et al*. Efficient estimation of word representations in vector space. *ArXiv13013781 Cs*; 2013.
32. Pennington J, Socher R, Manning CD. Glove: global vectors for word representation. In: *EMNLP*; 2014:1532–43. Accessed April 24, 2017; Doha, Qatar.
33. Iacobacci I, Pilehvar MT, Navigli R. Embeddings for word sense disambiguation: an evaluation study. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. San Diego, CA: Association for Computational Linguistics; 2016: 897–907; Berlin, Germany.
34. Pakhomov SVS, Finley G, McEwan R, *et al*. Corpus domain effects on distributional semantic modeling of medical terms. *Bioinformatics* 2016; 32: 3635–44. doi: 10.1093/bioinformatics/btw529
35. Kågebäck M, Salomonsson H. Word sense disambiguation using a bidirectional LSTM. *ArXiv160603568*; 2016; Osaka, Japan.
36. Pesaranghader A, Pesaranghader A, Matwin S, *et al*. One single deep bidirectional LSTM network for word sense disambiguation of text data. *ArXiv180209059*; 2018.
37. Bartunov S, Kondrashkin D, Osokin A, *et al*. Breaking sticks and ambiguities with adaptive skip-gram. In: *Artificial Intelligence and Statistics*. 2016. 130–8.
38. Pelevina M, Arefyev N, Biemann C, *et al*. Making sense of word embeddings. *ArXiv170803390 Cs*; 2017: 174–83.
39. Neelakantan A, Shankar J, Passos A, *et al*. Efficient non-parametric estimation of multiple embeddings per word in vector space. *ArXiv Prepr ArXiv150406654*; 2015:1059–69.
40. Chen X, Liu Z, Sun M. A unified model for word sense representation and disambiguation. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2014:1025–35; Doha, Qatar.
41. Panchenko A, Ruppert E, Faralli S, *et al*. Unsupervised does not mean uninterpretable: The case for word sense induction and disambiguation. In: *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. San Diego, CA: Association for Computational Linguistics; 2017: 86–98; Valencia, Spain.
42. Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput* 1997; 9 (8): 1735–80.
43. Graves A, Schmidhuber J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw* 2005; 18 (5–6): 602–10.

44. Sutskever I, Vinyals O, Le QV, Sequence to sequence learning with neural networks. In: Ghahramani Z, Welling M, Cortes C, *et al*., eds. *Advances in Neural Information Processing Systems 27*. Red Hook, NY: Curran Associates, Inc; 2014: 3104–12. Accessed April 4, 2017.
45. Raganato A, Bovi CD, Navigli R. Neural sequence learning models for word sense disambiguation. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 2017. 1156–67; Copenhagen, Denmark.
46. Ahmed M, Samee MR, Mercer RE. A novel neural sequence model with multiple attentions for word sense disambiguation. *ArXiv Prepr ArXiv180901074*; 2018.
47. Akata Z, Reed S, Walter D, *et al*. Evaluation of output embeddings for fine-grained image classification. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015: 2927–36; Boston, Massachusetts, USA.
48. Romera-Paredes B, Torr P. An embarrassingly simple approach to zero-shot learning. In: *International Conference on Machine Learning*. 2015: 2152–61; Lille, France.
49. Zhang L, Xiang T, Gong S. Learning a deep embedding model for zero-shot learning. *ArXiv Prepr ArXiv161105088*; 2017: 3010–9.
50. Kodirov E, Xiang T, Gong S. Semantic autoencoder for zero-shot learning. *ArXiv Prepr ArXiv170408345*; 2017; 4447–56; Miami, Florida, USA.
51. Liu Y, McInnes BT, Pedersen T, *et al*. Semantic relatedness study using second order co-occurrence vectors computed from biomedical corpora, UMLS and WordNet. In: *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*. New York, NY: ACM; 2012: 363–72.
52. Pesaranghader A, Matwin S, Sokolova M, *et al*. simDEF: definition-based semantic similarity measure of gene ontology terms for functional similarity analysis of genes. *Bioinformatics* 2016; 32 (9): 1380–7. doi: 10.1093/bioinformatics/btv755
53. Pesaranghader A, Pesaranghader A, Rezaei A, *et al*. Gene functional similarity analysis by definition-based semantic similarity measurement of GO terms. In: Sokolova M, van Beek P , eds. *Advances in Artificial Intelligence*. New York, NY: Springer International; 2014: 203–14.
54. Levy O, Goldberg Y. Neural word embedding as implicit matrix factorization. In: *Advances in Neural Information Processing Systems*. 2014: 2177–85; Montreal, Canada; Accessed April 24, 2017.
55. Baroni M, Dinu G, Kruszewski G. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. San Diego, CA: Association for Computational Linguistics; 2014: 238–47; Baltimore, Maryland, USA.
56. Pakhomov S, McInnes B, Adam T, *et al*. Semantic similarity and relatedness between clinical terms: an experimental study. *AMIA Annu Symp Proc* 2010; 2010: 572–6; Kuala Lumpur, Malaysia.
57. Pesaranghader A, Muthaiyah S, Pesaranghader A. Improving gloss vector semantic relatedness measure by integrating pointwise mutual information: Optimizing second-order co-occurrence vectors computed from biomedical corpus and UMLS. In: *2013 International Conference on Informatics and Creative Multimedia*. Pisataway, NJ: IEEE; 2013; 196–201; Seoul, South Korea.
58. Pesaranghader A, Rezaei A, Pesaranghader A. Adapting gloss vector semantic relatedness measure for semantic similarity estimation: an evaluation in the biomedical domain. In: Kim W, Ding Y, Kim H-G, eds. *Semantic Technology*. New York, NY: Springer International; 2014: 129–45.
59. Golub GH, Reinsch C. Singular value decomposition and least squares solutions. *Numer Math* 1970; 14 (5): 403–20.
60. Flekova L, Gurevych I. Supersense embeddings: A unified model for supersense interpretation, prediction, and utilization. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. San Diego, CA: Association for Computational Linguistics; 2016: 2029–41; Berlin, Germany.
61. McInnes BT, Pedersen T. Evaluating measures of semantic similarity and relatedness to disambiguate terms in biomedical text. *J Biomed Inform* 2013; 46 (6): 1116–24.
62. Pedersen T, Kolhatkar V. WordNet:: SenseRelate:: AllWords: a broad coverage word sense tagger that maximizes semantic relatedness. In: *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Demonstration Session*. San Diego, CA: Association for Computational Linguistics; 2009: 17–20.
63. Nair V, Hinton GE. Rectified linear units improve restricted Boltzmann machines. In: *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*; 2010: 807–14; Haifa, Israel; Accessed April 5, 2017.
64. Pesaranghader A, Pesaranghader A, Mustapha N. Word sense disambiguation for biomedical text mining using definition-based semantic relatedness and similarity measures. *Int J Biosci Biochem Bioinformatics* 2014; 4: 280.
65. Yang Z, Yang D, Dyer C, *et al*. Hierarchical attention networks for document classification. In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, CA: Association for Computational Linguistics; 2016: 1480–89.