Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

Subject Section

Cross-type Biomedical Named Entity Recognition with Deep Multi-Task Learning

Xuan Wang [1,*], Yu Zhang [1], Xiang Ren [2,*], Yuhao Zhang [3], Marinka Zitnik [4], Jingbo Shang [1], Curtis Langlotz [3] and Jiawei Han [1]

[1]Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA,
[2]Department of Computer Science, University of Southern California, Los Angeles, CA 90089, USA,
[3]School of Medicine, Stanford University, Stanford, CA 94305, USA, and
[4]Department of Computer Science, Stanford University, Stanford, CA 94305, USA

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Biomedical named entity recognition (BioNER) is the most fundamental task in biomedical text mining. State-of-the-art BioNER systems often require handcrafted features specifically designed for each type of biomedical entities. This feature generation process requires intensive labors from biomedical and linguistic experts, and makes it difficult to adapt these systems to new biomedical entity types. Although recent studies explored using neural network models for BioNER to free experts from manual feature generation, these models still require substantial human efforts to annotate massive training data.
Results: We propose a multi-task learning framework for BioNER that is based on neural network models to save human efforts. We build a global model by collectively training multiple models that share parameters, each model capturing the characteristics of a different biomedical entity type. In experiments on five BioNER benchmark datasets covering four major biomedical entity types, our model outperforms state-of-the-art systems and other neural network models by a large margin, even when only limited training data are available. Further analysis shows that the large performance gains come from sharing character- and word-level information between different biomedical entities. The approach creates new opportunities for text-mining approaches to help biomedical scientists better exploit knowledge in biomedical literature.
Availability: The source code for our models is available at https://github.com/yuzhimanhua/lm-lstm-crf, and the corpora are available at https://github.com/cambridgeltl/MTL-Bioinformatics-2016.
Contact: xwang174@illinois.edu, xiangren@usc.edu

1 Introduction

Biomedical text mining is an important tool to support large-scale biomedical data analysis, such as biomedical network construction [Zhou *et al.*, 2014], gene prioritization [Aerts *et al.*, 2006, Liu *et al.*, 2015], drug repositioning [Wang and Zhang, 2013], finding literature support of experimental findings [Morris *et al.*, 2012, Willer *et al.*, 2013, Fehrmann *et al.*, 2015], generating hypothesis [Zhou *et al.*, 2014, Al-Aamri *et al.*, 2017, Rastegar-Mojarad *et al.*, 2015] and database curation [Li *et al.*, 2015]. The most fundamental task in biomedical text mining is biomedical named entity recognition (BioNER) that automatically recognizes and extracts biomedical entities (e.g., genes, proteins, chemicals and diseases) from text [Jensen *et al.*, 2006, Rebholz-Schuhmann *et al.*, 2012]. BioNER can be used to identify new gene names from text [Smith *et al.*, 2008]. It also serves as a primitive step of many downstream applications, such as relation extraction [Cokol *et al.*, 2005] and knowledge base completion [Dai *et al.*, 2010, Gonzalez *et al.*, 2015, Szklarczyk *et al.*, 2017, Wei *et al.*, 2013, Xie *et al.*, 2013, Szklarczyk *et al.*, 2015].

BioNER is most commonly approached as a sequence labeling problem to sequentially assign a label to each word in a sentence. State-of-the-art BioNER systems often require handcrafted features (e.g., capitalization, prefix and suffix) to be specifically designed for each type of biomedical entities [Sondhi, 2008, Ando, 2007, Leaman *et al.*, 2015, Leaman and Lu, 2016, Zhou and Su, 2004]. This feature generation process

requires intensive labors from biomedical and linguistic experts [Leser and Hakenberg, 2005], and makes it difficult to adapt these systems to recognize new biomedical entity types. Moreover, the accuracy of BioNER tools is still a limiting factor for current biomedical text mining pipelines [Huang and Lu, 2015].

Recent studies explored using neural network models for BioNER to automatically generate quality features. For example, Crichton *et al.*, 2017 took each word token and its surrounding context words as input into a convolutional neural network (CNN). Habibi *et al.*, 2017 adopted the model from Lample *et al.*, 2016 and took word embeddings as input into a bidirectional long short-term memory-conditional random field (BiLSTM-CRF) model. These neural network models free experts from manual feature generation, but still require human efforts for training data annotation. Large training data sets are preferred but currently unavailable for the neural network models that have a large number of parameters, which has limited the performance of these models. Although the neural network models show an improved performance compared with strong baselines (e.g., CRF models [Lafferty *et al.*, 2001]), they still cannot outperform state-of-the-art systems that utilize handcrafted features.

To achieve a better performance using the limited available training data, some recent studies explored using multi-task learning (MTL) with neural network models. The key idea of MTL is to collectively train several related tasks at the same time, so that each task will benefit from the existing annotations of the other tasks to save human efforts. MTL can reach a globally optimized performance on all the tasks with limited available training data for each single task. It has been successfully applied to several tasks such as natural language processing [Collobert and Weston, 2008], speech recognition [Deng *et al.*, 2013], computer vision [Girshick, 2015] and drug discovery [Ramsundar *et al.*, 2015]. MTL has also been applied to BioNER with limited success. Crichton *et al.*, 2017 incorporated MTL with CNN for BioNER. However, their CNN model is not as efficient as recently proposed BiLSTM models for BioNER. The model takes word-level features as input, and does not capture the character-level lexical information. This multi-task CNN model still cannot outperform state-of-the-art systems that utilize handcrafted features.

In this paper, we propose a new multi-task learning framework based on neural networks for BioNER. The proposed framework frees biomedical experts from manual feature generation while also achieving excellent performance using limited available training data. Our multi-task model is built upon a single-task neural network model [Liu *et al.*, 2017a], a BiLSTM-CRF model with an additional context-dependent BiLSTM layer for character embedding. A prominent property of our model is that inputs from different datasets can share both character- and word-level information. The information sharing is achieved by re-using the same parameters in the BiLSTM units. More specifically, in the multi-task setting, input sentences from different datasets go through the same neural network model. The model outputs the original sentences labeled with entity types (Figure 1). We compare the proposed multi-task model with state-of-the-art BioNER systems [Sondhi, 2008, Ando, 2007, Leaman *et al.*, 2015, Leaman and Lu, 2016, Zhou and Su, 2004] and neural network NER tools [Crichton *et al.*, 2017, Habibi *et al.*, 2017, Ma and Hovy, 2016, Lample *et al.*, 2016, Liu *et al.*, 2017a] on five BioNER benchmark datasets covering four major entity types. Results show that the proposed model achieves substantially better performance than state-of-the-art BioNER systems. Altogether, this work introduces a text-mining approach that can help scientists exploit knowledge buried in the vast biomedical literature in a systematic and unbiased way.

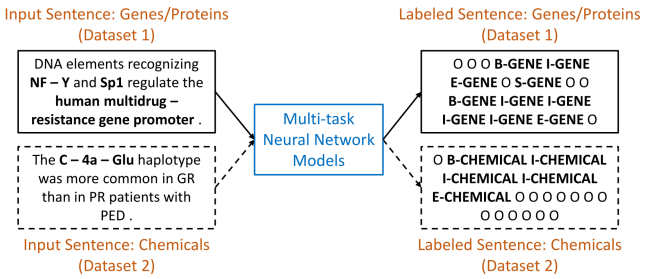

Fig. 1. The illustrative figure of neural network based multi-task framework. The input are sentences from different biomedical datasets. Each sentence will go through the same multi-task neural network models in the center and update the same set of parameters. Then the model will output the entity type labels for each word in the input sentences, such as genes and chemicals, using the BIOES schema.

2 Background

In this section, we introduce basic neural network architectures that are relevant to our multi-task learning framework.

2.1 Long Short-Term Memory (LSTM)

Long short-term memory neural network is a specific type of recurrent neural network that models dependencies between elements in a sequence through recurrent connections. The input to an LSTM network is a sequence of vectors $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_T\}$, where vector $\boldsymbol{x}_i$ is a representation vector of a word in the input sentence. The output is a sequence of vectors $\boldsymbol{H} = \{\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_T\}$, where $\boldsymbol{h}_i$ is a hidden state vector. At step t of the recurrent calculation, the network takes $\boldsymbol{x}_t, \boldsymbol{c}_{t-1}, \boldsymbol{h}_{t-1}$ as inputs and produces $\boldsymbol{c}_t, \boldsymbol{h}_t$ via the following intermediate calculations:

$$\boldsymbol{i}_t = \sigma(\boldsymbol{W}^i \boldsymbol{x}_t + \boldsymbol{U}^i \boldsymbol{h}_{t-1} + \boldsymbol{b}^i) \quad (1)$$

$$\boldsymbol{f}_t = \sigma(\boldsymbol{W}^f \boldsymbol{x}_t + \boldsymbol{U}^f \boldsymbol{h}_{t-1} + \boldsymbol{b}^f) \quad (2)$$

$$\boldsymbol{o}_t = \sigma(\boldsymbol{W}^o \boldsymbol{x}_t + \boldsymbol{U}^o \boldsymbol{h}_{t-1} + \boldsymbol{b}^o) \quad (3)$$

$$\boldsymbol{g}_t = tanh(\boldsymbol{W}^g \boldsymbol{x}_t + \boldsymbol{U}^g \boldsymbol{h}_{t-1} + \boldsymbol{b}^g) \quad (4)$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \boldsymbol{g}_t \quad (5)$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot tanh(\boldsymbol{c}_t), \quad (6)$$

where $\sigma(\cdot)$ and $tanh(\cdot)$ denote element-wise sigmoid and hyperbolic tangent functions, respectively, and $\odot$ denotes element-wise multiplication. The $\boldsymbol{i}_t$, $\boldsymbol{f}_t$ and $\boldsymbol{o}_t$ are referred to as input, forget, and output gates, respectively. At $t = 1$, $\boldsymbol{h}_0$ and $\boldsymbol{c}_0$ are initialized to zero vectors. The trainable parameters are $\boldsymbol{W}^j$,$\boldsymbol{U}^j$ and $\boldsymbol{b}^j$ for $j \in \{i, f, o, g\}$.

The LSTM architecture described above can only process the input in one direction. The bi-directional long short-term memory (BiLSTM) model improves the LSTM by feeding the input to the LSTM network twice, once in the original direction and once in the reversed direction. Outputs from both directions are concatenated to represent the final output. This design allows for detection of dependencies from both previous and subsequent words in a sequence.

2.2 Bi-directional Long Short-Term Memory-Conditional Random Field (BiLSTM-CRF)

A naive way of applying the BiLSTM network to sequence labeling is to use the output hidden state vectors to make independent tagging decisions. However, in many sequence labeling tasks such as BioNER, it is useful to also model the dependencies across output tags. The

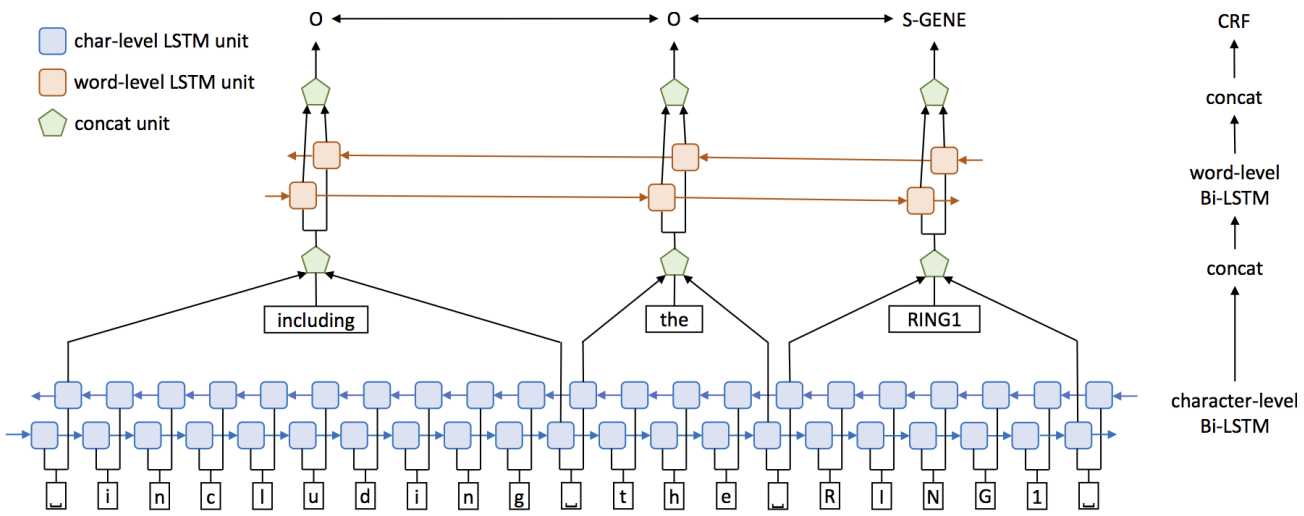

Fig. 2. Single-task learning neural network architecture. The input is a sentence from the biomedical literature. The white rectangles represent character and word embeddings. The blue rectangles represent the first character-level BiLSTM. The red rectangles represent the second word-level BiLSTM. The green pentagons represent the concatenation units. The tags on the top, e.g., 'O', 'S-GENE', are the output of the final CRF layer, which are the entity labels we get for each word in the sentence.

BiLSTM-CRF network adds a conditional random field (CRF) layer on top of a BiLSTM network. This BiLSTM-CRF network takes the input sequence $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n\}$ to predict an output label sequence $\boldsymbol{y} = \{y_1, y_2, ..., y_n\}$. A score is defined as:

$$s(\boldsymbol{X}, \boldsymbol{y}) = \sum_{i=0}^{n} \boldsymbol{A}_{y_i, y_{i+1}} + \sum_{i=1}^{n} \boldsymbol{P}_{i, y_i}, \tag{7}$$

where $\boldsymbol{P}$ is an $n \times k$ matrix of the output from the BiLSTM layer, n is the sequence length, k is the number of distinct labels, $\boldsymbol{A}$ is a $(k+2) \times (k+2)$ transition matrix and $\boldsymbol{A}_{i,j}$ represents the transition probability from the i-th label to the j-th label. Note that two additional labels <*start*> and <*end*> are used to represent the start and end of a sentence, respectively. We further define $\boldsymbol{Y}_{\boldsymbol{X}}$ as all possible sequence labels given the input sequence $\boldsymbol{X}$. The training process maximizes the log-probability of the label sequence $\boldsymbol{y}$ given the input sequence $\boldsymbol{X}$:

$$log(p(\boldsymbol{y}|\boldsymbol{X})) = log \frac{e^{s(\boldsymbol{X}, \boldsymbol{y})}}{\sum_{\boldsymbol{y}' \in \boldsymbol{Y}_{\boldsymbol{X}}} e^{s(\boldsymbol{X}, \boldsymbol{y}')}}. \tag{8}$$

A three-layer BiLSTM-CRF architecture is employed by Lample *et al.*, 2016 and Habibi *et al.*, 2017 to jointly model the word and the character sequences in the input sentence. In this architecture, the first BiLSTM layer takes character embedding sequence of each word as input, and produces a character-level representation vector for this word as output. This character-level vector is then concatenated with a word embedding vector, and fed into a second BiLSTM layer. Lastly, a CRF layer takes the output vectors from the second BiLSTM layer, and outputs the best tag sequence by maximizing the log-probability in Equation 8.

In practice, the character embedding vectors are randomly initialized and co-trained during the model training process. The word embedding vectors are retrieved directly from a pre-trained word embedding lookup table. The classical Viterbi algorithm is used to infer the final labels for the CRF model. The three-layer BiLSTM-CRF model is a differentiable neural network architecture that can be trained by backpropagation.

3 Deep multi-task learning framework

In this section, we first introduce the baseline single-task model [Liu *et al.*, 2017a]. Then we introduce three proposed multi-task models built on top of the single-task model.

3.1 Baseline single-task model (STM)

The vanilla BiLSTM-CRF model can learn high-quality representations for words that appeared in the training dataset. However, it often fails to generalize to out-of-vocabulary words, i.e., words that did not appear in the training dataset. These out-of-vocabulary words are especially common in biomedical text. Therefore, for the baseline single-task BioNER model, we use a neural network architecture that better handles out-of-vocabulary words. As shown in Figure 2, our single-task model consists of three layers. In the first layer, a BiLSTM network is used to model the character sequence of the input sentence. We use character embedding vectors as input to the network. Hidden state vectors at the word boundaries of this character-level BiLSTM are then selected and concatenated with word embedding vectors to form word representations. Next, these word representation vectors are fed into a second word-level BiLSTM layer. Lastly, output of this word-level BiLSTM is fed into the a CRF layer for label prediction. Compared to the vanilla BiLSTM-CRF model, a major advantage of this model is that it can infer the meaning of an out-of-vocabulary word from its character sequence and other characters around it. For example, the network is able to infer that "RING2" is likely to represent a gene symbol, even though then network may have only seen the word "RING1" during training.

3.2 Multi-task models (MTMs)

An important characteristic of the BioNER task is the limited availability of supervised training data for each entity type. We propose a multi-task learning approach to address this problem by training different BioNER models on datasets with different entity types while sharing parameters across these models. We hypothesis that the proposed approach can make more efficient use of the data and encourage the models to learn representations that better generalize.

We give a formal definition of the multi-task setting as the following. Given m datasets, for $i \in \{1, ..., m\}$, each dataset D_i consists of n_i

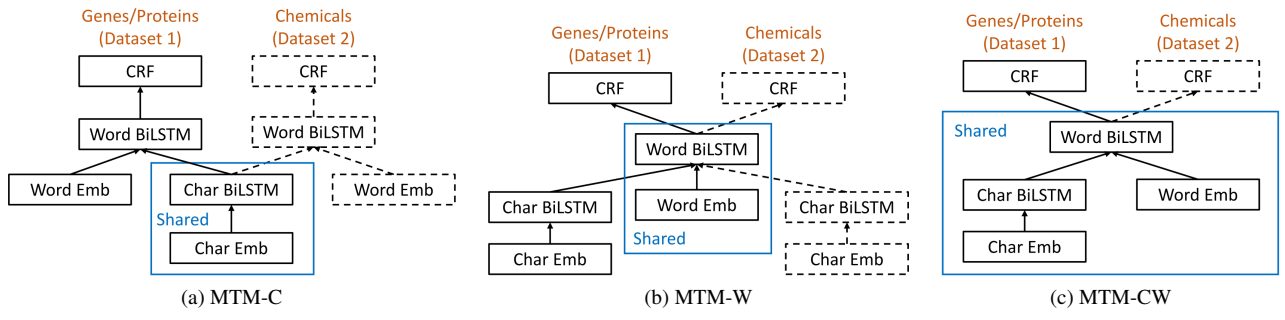

Fig. 3. Three multi-task learning neural network models. (a) MTM-C: multi-task learning neural network with a shared character layer and a task-specific word layer, (b): MTM-W: multi-task learning neural network with a task-specific character layer and a shared word layer, (c) MTM-CW: multi-task learning neural network with shared character and word layers.

training samples, i.e., $D_i = \{\boldsymbol{x}_j^i, y_j^i\}_{j=1}^{n_i}$. We denote the training matrix for each dataset as $\boldsymbol{X}^i = \{\boldsymbol{x}_1^i, ..., \boldsymbol{x}_{n_i}^i\}$ and the labels for each dataset as $\boldsymbol{y}^i = \{y_1^i, ..., y_{n_i}^i\}$. A multi-task model therefore consists of m different models, each trained on a separate dataset, while sharing part of the model parameters across datasets. The loss function L is:

$$L = \sum_{i=1}^{m} \lambda_i L_i = \sum_{i=1}^{m} \lambda_i log(p(\boldsymbol{y}^i | \boldsymbol{X}^i)). \quad (9)$$

The log-likelihood term is shown in Equation 8 and λ_i is a positive regularization parameter that controls the contribution of each dataset. In the experiments, we set $\lambda_i = 1$ for $i \in \{1, ..., m\}$.

We propose three different multi-task models, as illustrated in Figure 3. These three models differ in which part of the model parameters are shared across multiple datasets:

MTM-C This model shares the character-level parameters among tasks but uses task-specific word-level parameters for each task. All datasets are iteratively used to train the model. When a dataset is used, the parameters updated during the training are the word-level parameters specific to this task and the shared character-level parameters. The detailed architecture of this multi-task model is shown in Figure 3(a).

MTM-W This model uses task-specific character-level parameters for each task but shares the word-level parameters among tasks. When a dataset is used, the parameters updated during the training are the character-level parameters specific to this task and the shared word-level parameters. The detailed architecture of this multi-task model is shown in Figure 3(b).

MTM-CW This model shares both character- and word-level parameters among tasks. During the training, all datasets share and update the same set of parameters at both the character level and the word level. Each dataset has its specific CRF layer for label prediction. MTM-CW is the most comprehensive among the three proposed multi-task models. It enables sharing both character- and word-level information between different biomedical entities, while the other two models only enable sharing part of the information. The detailed architecture of this multi-task model is shown in Figure 3(c).

4 Experimental setup

In this section, we introduce the datasets, evaluation metrics, pre-trained word embeddings, and model training details in the experiments.

4.1 Datasets

We use five BioNER datasets collected by Crichton *et al.*, 2017. These five datasets cover major biomedical entities (e.g., genes, proteins, chemicals, diseases) and include state-of-the-art performance on each dataset for comparison. Each dataset consists of three parts: a training set (∼60% of the samples) for model training, a development set (∼10% of the samples) for model tuning and a test set (∼30% of the samples) for evaluation. During preprocessing, word labels are encoded using an IOBES scheme. In this scheme, for example, a word describing a gene entity is tagged with "B-Gene" if it is at the beginning of the entity, "I-Gene" if it is in the middle of the entity, and "E-Gene" if it is at the end of the entity. Single-word gene entities are tagged with "S-Gene". All other words that do not describe any specific entities are tagged as 'O'. All datasets are publicly available and can be downloaded from https://github.com/cambridgeltl/MTL-Bioinformatics-2016. Detailed statistics of each dataset are listed in Table 1. We also include a short description of each dataset and the corresponding state-of-the-art system below.

BC2GM The state-of-the-art system in the BioCreative II gene mention recognition task is a semi-supervised learning method using alternating structure optimization [Ando, 2007].

BC4CHEMD The state-of-the-art system in the BioCreative IV chemical entity mention recognition task is the *tmChem* system, which uses an ensemble model consisting of two CRF classifiers [Leaman *et al.*, 2015].

BC5CDR The state-of-the-art system in the most recent BioCreative V chemical and disease mention recognition task is the *TaggerOne* system, which uses a semi-Markov model for joint entity recognition and normalization [Leaman and Lu, 2016] .

NCBI-Disease The NCBI disease corpus was initially introduced for disease name recognition and normalization. It has been widely used for a lot of applications. The state-of-the-art system on this dataset [Leaman and Lu, 2016] is also the *TaggerOne* system.

JNLPBA The state-of-the-art system [Zhou and Su, 2004] for the 2004 JNLPBA shared task on biomedical entity recognition uses a hidden markov model (HMM). Although this task and the model is a bit old compared with the others, it still remains a competitive benchmark method for comparison.

4.2 Evaluation metrics

We use the development set of each dataset for model tuning and report the performance on the test set. We deem each predicted entity as correct only if it is an *exact match* to an entity at the same position in the ground truth annotation. Then we calculate the precision, recall and F1 scores on all datasets and all entity types. For error analysis, we compare the number of false positive (FP) and false negative (FN) labels in the single-task and the multi-task models.

Table 1. Information of the five biomedical NER datasets used in the experiments.

Dataset	Size	Entity types and counts
BC2GM	20,000 sentences	Gene/Protein (24,583)
BC4CHEMD	10,000 abstracts	Chemical (84,310)
BC5CDR	1,500 articles	Chemical (15,935), Disease (12,852)
NCBI-Disease	793 abstracts	Disease (6,881)
JNLPBA	2,404 abstracts	Gene/Protein (35,336), Cell Type (8,649), Cell Line (4,330), DNA (10,589), RNA (1,069)

The test set of the BC2GM dataset is constructed slightly differently compared to the test sets of other datasets. BC2GM additionally provides a list of alternative answers for each entity in the test set. A predicted entity is deemed correct as long as it matches the ground truth or one of the alternative answers. We refer to this measurement as *alternative match* and report scores under both *exact match* and *alternative match* for the BC2GM dataset.

4.3 Pre-trained word embeddings

We initialize the word embedding matrix with pre-trained word vectors from Pyysalo *et al.*, 2013 in all experiments. These word vectors are trained using the skip-gram model, as described in Mikolov *et al.*, 2013, for learning distributed representations of words using contextual information. Three sets of word vectors are provided with different training data. The first one is trained on the whole PubMed abstracts, the second one is trained on the PubMed abstracts together with all the full-text articles from PubMed Central (PMC), and the last one is different from the second one by also training the vectors on the Wikipedia corpora. We use the third set of word vectors that are trained on the abstracts from PubMed, full articles from PMC and the Wikipedia corpora for the model development. We also compare the performance with those obtained using the other two sets of word vectors. In all five datasets, rare words (i.e., words with frequency less than 5) are replaced by a special <*UNK*> token, whose embedding is randomly initialized and fine-tuned during model training.

4.4 Training details

To train the proposed single-task and multi-task models, we use a learning rate of 0.01 with a decay rate of 0.05 after every epoch of training. The dimension of word and character embedding vectors are set to be 200 and 30, respectively. We adopt a hidden state size of 200 for both character- and word-level BiLSTM layers. Note that in the original paper of Liu *et al.*, 2017a, they also incorporate advanced strategies such as highway structures to further improve the NER performance. We tested these variations but did not observe any significant boost in the performance. Therefore, we do not adopt these strategies in this work.

To compare the performance with other neural network models on each dataset, we directly reported the best results from Crichton *et al.*, 2017 and Habibi *et al.*, 2017. To compare with the model proposed by Ma and Hovy, 2016, we trained their model on the five datasets with the default parameter settings as used in their paper.

5 Results

5.1 Model performance comparison on benchmark datasets

We compare the proposed single-task (Section 3.1) and multi-task models (Section 3.2) with state-of-the-art BioNER systems and three neural network models from Crichton *et al.*, 2017, Lample *et al.*, 2016, Habibi *et al.*, 2017, and Ma and Hovy, 2016. The results (precision, recall and F1) are shown in Table 2. We measure statistical significance through a two-tailed t-test computed on the F1 scores in all reported experiments.

We observe that the MTM-CW model performs significantly better than state-of-the-art systems (column Benchmark dataset in Table 2) on three out of five datasets. On the remaining BC4CHEMD and BC5CDR datasets, MTM-CW achieves competitive performance. Following established practice in the literature, we use exact matching to compare benchmark performance on all the datasets except for the BC2GM, where we report benchmark performance based on alternative matching. The details of state-of-the-art systems and evaluation methods are described in Section 4.1 and 4.2, respectively. Furthermore, MTM-CW performs significantly better than other neural network models on all five datasets. These results show that the proposed multi-task learning neural network significantly outperforms state-of-the-art systems and other BioNER neural networks. In particular, the MTM-CW model consistently achieves a better performance than the single task model, demonstrating that multi-task learning is able to successfully leverage information across BioNER tasks and mutually enhance performance on every task. We further investigate the performance of three multi-task models (MTM-C, MTM-W, and MTM-CW, Table 3). Results show that the best performing multi-task model is MTM-CW, indicating the importance of lexical features in character-level BiLSTM as well as semantic features in word-level BiLSTM.

5.2 Effect of dataset characteristics on multi-task learning

To investigate how the dataset characteristics affect the performance of multi-task learning, we perform pairwise multi-task training with the MTM-CW model. For example, we train the model using BC2GM and each of the other four datasets to find the one that pairs best with BC2GM. Results are shown in Table 4.

Two important factors can influence collaboration between tasks. One is related to the sharing of entity types occurring in multiple datasets. For example, if two datasets have an entity type in common, they are more likely to become best partners. The other factor is the size of the partner dataset. If a dataset is relatively large in size, it is more likely to contain more sharable information and to become the best partner of another dataset. The results match our expectations. The best partner of BC5CDR (chemicals, diseases) is BC4CHEMD (chemicals) that shares a common entity type (chemicals) with BC5CDR and is also the largest among the four considered datasets. The best partner for BC4CHEMD (chemicals) is JNLPBA (gene/protein, DNA, cell type, cell line, RNA), which is the largest dataset considered in the study. Although JNLPBA does not have any entity type in common with BC4CHEMD, it likely contains the largest amount of potentially relevant biomedical information, and thus it is considered the best partnering dataset for BC4CHEMD. These results indicate that it is more likely to achieve a better performance in multi-task learning to pair datasets that contain common entity types or datasets that are large in size.

Table 2. Performances of baseline neural network models and the MTM-CW model. Significance test is performed on the F1 values. Bold: best scores, *: significantly worse than the MTM-CW model ($p \leq 0.05$), **: significantly worse than the MTM-CW model ($p \leq 0.01$).

		Dataset Benchmark	Crichton *et al.*	Lample *et al.* Habibi *et al.*	Ma and Hovy	Liu *et al.* STM	MTM-CW
BC2GM (Exact)	Precision	-	-	78.99	83.33	83.07	**83.98**
	Recall	-	-	78.16	81.25	82.02	**82.32**
	F1	-	73.17**	78.57**	82.28**	82.54*	**83.14**
BC2GM (Alternative)	Precision	88.48	-	86.11	83.50	88.21	**89.45**
	Recall	85.97	-	86.96	87.13	87.43	**88.67**
	F1	87.21**	84.41**	86.53**	85.27**	87.82*	**89.06**
BC4CHEMD	Precision	89.09	-	87.83	**90.59**	89.55	90.51
	Recall	85.75	-	85.45	82.63	84.62	**86.18**
	F1	87.39	83.02**	86.62*	86.43*	87.01*	**88.29**
BC5CDR	Precision	**89.21**	-	86.82	88.24	87.41	87.69
	Recall	84.45	-	86.40	78.79	83.05	**87.17**
	F1	86.76	83.90**	86.61*	83.24**	85.18**	**87.43**
NCBI-Disease	Precision	85.10	-	**86.43**	84.33	84.84	85.00
	Recall	80.80	-	82.92	83.77	85.39	**87.80**
	F1	82.90**	80.37**	84.64**	84.04**	85.10**	**86.37**
JNLPBA	Precision	69.42	-	71.35	**72.88**	72.29	72.72
	Recall	75.99	-	75.74	75.98	77.25	**77.83**
	F1	72.55**	70.09**	73.48**	74.40*	74.69*	**75.19**

Table 3. F1 scores of the different multi-task models. Bold: best scores, *: significantly worse than the MTM-CW model ($p \leq 0.05$), **: significantly worse than the MTM-CW model ($p \leq 0.01$).

Dataset	MTM-C	MTM-W	MTM-CW
BC2GM	80.81**	82.61*	**83.14**
BC4CHEMD	87.22*	87.69	**88.29**
BC5CDR	85.52**	86.52*	**87.43**
NCBI-Disease	84.48**	84.79**	**86.37**
JNLPBA	74.56*	74.70*	**75.19**

Table 4. F1 scores of the multi-task model for pairwise datasets. Refer to Table 1 for the entity types of each dataset. *: the performance of the best partner is significantly better than that of the second best partner ($p \leq 0.05$), **: the performance of the best partner is significantly better than that of the second best partner ($p \leq 0.01$).

Dataset	STM	Best Pairwise	Best Partner
BC2GM	82.54	82.88	BC4CHEMD
BC4CHEMD	87.01	87.85	JNLPBA*
BC5CDR	85.18	87.38	BC4CHEMD**
NCBI-Disease	85.10	85.19	BC4CHEMD
JNLPBA	74.69	74.97	BC2GM

5.3 Model performance comparison on major entities

We compared the performance of all models on the benchmark datasets in Section 5.1. Now we compare all models on four major biomedical entity types: genes/proteins, chemicals, diseases and cell lines. Each entity type comes from multiple datasets: genes/proteins from BC2GM and JNLPBA, chemicals from BC4CHEMD and BC5CDR, diseases from BC5CDR and NCBI-Disease, and cell lines from JNLPBA. The results of macro-F1 scores are shown in Figure 4.

The MTM-CW model performs consistently the best on all entity types compared with state-of-the-art systems (benchmark) and neural network models (Habibi *et al.*, 2017). The model from Habibi *et al.*, 2017 is generally better than state-of-the-art systems except on the chemical entities. These results further confirm that the multi-task neural network model achieves a significantly better performance compared with state-of-art systems and neural network models for BioNER.

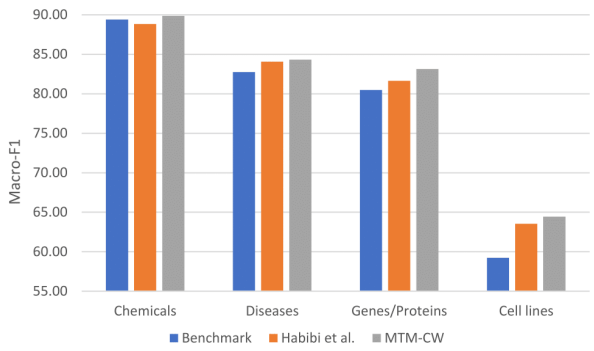

Fig. 4. Macro-F1 scores of the multi-task model compared with benchmark on different entities. Benchmark refers to the performance of state-of-the-art BioNER systems.

5.4 Integration of biomedical entity dictionaries

Biomedical entity dictionary is a list of entity names that belong to a specific biomedical entity type. It is another rich resource for BioNER, additional to the labeled training data. We retrieve three biomedical entity dictionaries, i.e., genes/proteins, chemicals and diseases, from the comparative toxicogenomics database (CTD) [Davis *et al.*, 2017]. Then we incorporate the dictionary information into the neural network models in two ways: (1) dictionary post-processing to match the 'O'-labeled entities with the dictionary to reduce the false negative rate, (2) dictionary pre-processing to add extra dimensions as part of the input into the word-level BiLSTM. The added dimensions represent whether a word sequence consisting of the word and its consecutive neighbors is in the dictionary. The length of the word sequence is limited to six words, thus 21 dimensions are added for each entity type. We compare the performance of MTM-CW with and without added dictionaries. The results are shown in Table 5.

No significant improvement in performance is observed when biomedical entity dictionaries are included into the MTM-CW model at

Table 5. F1 scores of the multi-task model with CTD biomedical entity dictionaries. Bold: best scores, *: significantly worse than the MTM-CW model ($p \leq 0.05$), **: significantly worse than the MTM-CW model ($p \leq 0.01$).

Dataset	MTM-CW	+Dictionary Pre-process	+Dictionary Post-process
BC2GM	**83.14**	82.96	75.38**
BC4CHEMD	**88.29**	88.25	86.99*
BC5CDR	87.43	**87.58**	84.79**
NCBI-Disease	**86.37**	86.29	85.41*
JNLPBA	**75.19**	74.91	72.31**

Table 6. Error analysis of false positive (FP) and false negative (FN) ratios (%) among all the wrongly labeled entities of the single-task and multi-task models.

Type		STM	MTM-CW
BC2GM	FP	40.74	41.57
	FN	43.06	36.41
BC4CHEMD	FP	36.21	54.57
	FN	50.00	30.30
BC5CDR	FP	38.31	48.36
	FN	44.93	33.93
NCBI-Disease	FP	50.45	35.26
	FN	31.25	49.36
JNLPBA	FP	34.12	29.99
	FN	34.76	36.24

the pre-processing stage. Moreover, including dictionaries at the post-processing stage even reduces the performance leading to increased false positive rate. These results indicate that multi-task model can learn an excellent representation using only labeled training data and then generalize it to previously unseen test data. As a result, integrating additional signal into the training process in the form of entity dictionaries does not lead to an obvious performance gain.

5.5 False error analysis

To better understand the performance improvement of multi-task models over single-task models, we investigate false positive and false negative ratios among all the wrongly labeled entities of the single-task and multi-task models. Since sequence labeling generates partially labeled entities, we only count completely missed entities as false negative and completely wrongly labeled entities as false positive labels. The results are shown in Table 6.

Across five datasets, the single-task model has a higher ratio of false negatives than false positives, while the opposite holds true for the multi-task model. These results indicate that the major improvement of multi-task models comes from the decrease in false negative labels, an appealing property achieved by collectively learning features for multiple entity types across different datasets. However, a major problem of learning from the others is that the learned features can be noisy for the target entity recognition, which increases the chance of false positive labeling.

6 Discussion

6.1 Impact of word embeddings

The quality of the pre-trained word embeddings, i.e., how well they capture the semantic similarity between different words, can lead to a great difference in the model performance. We compare three sets of word embeddings in the best-performing MTM-CW model. The three sets of word embeddings include word embeddings trained with PubMed abstracts, PubMed abstracts + PMC full papers and PubMed abstracts + PMC full papers + Wikipedia corpus. The F1 scores are shown in Figure 5.

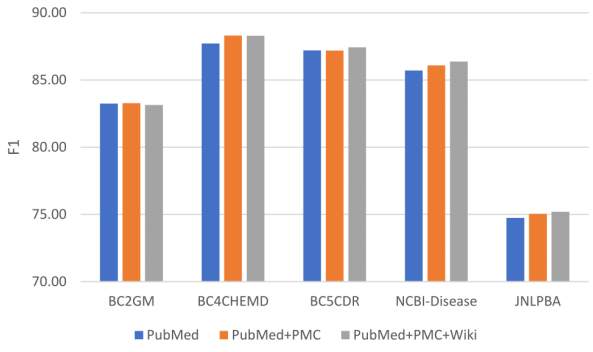

Fig. 5. F1 scores with pre-trained word embeddings from the Pubmed, Pubmed+PMC and Pubmed+PMC+Wikipedia corpus.

The word embeddings of PubMed + PMC + Wikipedia generally have the best performance compared with the other two. We used this set of word embeddings for the model development. However, the other two sets of word embeddings have a better performance in the BC2GM dataset, possibly because the gene entities of the BC2GM dataset contain some letter abbreviations. These abbreviations could be confused with some common words. For example, “was” is a gene symbol, which is also a common word that frequently appears in the general corpus. This word ambiguity could lead to a decreased quality of learned word embeddings for gene entities when trained with a large general corpus such as Wikipedia.

6.2 Case study

To investigate the major advantages of the multi-task models compared with the single task models, we examine some sentences with predicted labels shown in Table 7. The true labels and the predicted labels of each model are underlined in a sentence.

One major challenge of BioNER is to recognize a long entity with integrity, especially for recognizing gene/protein and chemical entities. In example 1 in Table 7, the true gene entity is “endo-beta-1,4-glucanase-encoding genes”. The single-task model tends to break this whole entity into two parts separated by a comma, which is a common practice for short entity recognition in general NER. In contrast, the multi-task model can detect this gene entity as a whole. We believe this result is due to co-training of the model with multiple datasets containing chemical entities and learning features from long chemical entity training examples.

Another challenge is to detect the correct boundaries of biomedical entities. For right boundary detection, one common mistake is to include non-entity tokens as part of the true entity. In example 2 in Table 7, the correct protein entity is “SMase” in the phrase “SMase - sphingomyelin complex structure”. The single-task models recognize the whole phrase “SMase - sphingomyelin complex structure” as a protein entity. However, the word “sphingomyelin” is actually a kind of lipid but not a protein in the phrase, thus should not be labeled as a protein. Our multi-task model is able to detect the correct part as a protein, probably also due to seeing more examples from other datasets which may contain “sphingomyelin” as a non-chemical entity. For the left boundary detection, one common mistake is to miss some adjective words as part of the true entity. It is common for disease entity recognition. For example, in example 4, the adjective words “human” and “complement factor” in front of “H deficiency” should be included as part of the true entity. The single-task models missed the adjective words and detected the entity as “H deficiency” or “complement factor H deficiency”, while the multi-task model is able to detect the correct right boundary of the entity in this sentence.

Table 7. Case study of the prediction results. It shows the advantage of the multi-task neural network model compared with the baseline model and single-task model. The true labels and the predicted labels of each model are underlined in the sentence.

		Genes/Proteins
Case 1	True label	This fragment contains two complete endo - beta - 1, 4 - glucanase - encoding genes, designated celCCC and celCCG.
	Habibi	This fragment contains two complete endo - beta - 1, 4 - glucanase - encoding genes, designated celCCC and celCCG.
	STM	This fragment contains two complete endo - beta - 1, 4 - glucanase - encoding genes, designated celCCC and celCCG.
	MTM-CW	This fragment contains two complete endo - beta - 1, 4 - glucanase - encoding genes, designated celCCC and celCCG.
Error		Entity integrity: break a long entity into parts and lose the entity integrity.
Case 2	True label	A model for the SMase - sphingomyelin complex structure was built to investigate how the SMase specifically recognizes its substrate.
	Habibi	A model for the SMase - sphingomyelin complex structure was built to investigate how the SMase specifically recognizes its substrate.
	STM	A model for the SMase - sphingomyelin complex structure was built to investigate how the SMase specifically recognizes its substrate.
	MTM-CW	A model for the SMase - sphingomyelin complex structure was built to investigate how the SMase specifically recognizes its substrate.
Error		Right boundary error: false detection of non-entity tokens as part of the true entity.
		Chemicals
Case 3	True label	Cyclo - (His, Leu): a new microbial diketopiperazine from a terrestrial Bacillus subtilis strain B38.
	Habibi	Cyclo - (His, Leu): a new microbial diketopiperazine from a terrestrial Bacillus subtilis strain B38.
	STM	Cyclo - (His, Leu): a new microbial diketopiperazine from a terrestrial Bacillus subtilis strain B38.
	MTM-CW	Cyclo - (His, Leu): a new microbial diketopiperazine from a terrestrial Bacillus subtilis strain B38.
Error		Entity integrity: break a long entity into parts and lose the entity integrity.
		Diseases
Case 4	True label	... human complement factor H deficiency associated with hemolytic uremic syndrome.
	Habibi	...human complement factor H deficiency associated with hemolytic uremic syndrome.
	STM	... human complement factor H deficiency associated with hemolytic uremic syndrome.
	MTM-CW	... human complement factor H deficiency associated with hemolytic uremic syndrome.
Error		Left boundary error: fail to detect the correct left boundary of the true entity due to some adjective words in front.

In summary, the multi-task model works better at dealing with two critical challenges for BioNER: (1) recognizing long entities with integrity for genes, proteins and chemicals, and (2) detecting correct right boundaries with appropriate adjective words in front of disease entities. Both improvements come from collectively training multiple datasets with different entity types and sharing useful information between datasets.

6.3 Related work

Neural networks are gaining great popularity in sequence labeling tasks such as named entity recognition in the past few years. In the general domain, several approaches have been proposed using the BiLSTM-CRF architectures. For example, Huang *et al.*, 2015 used a BiLSTM for word-level representation and a stacked CRF layer for label prediction. However, they did not use any CNN or RNN to encode character-level information. Instead, they still use the traditional handcrafted features. Chiu and Nichols, 2016 proposed to use CNN and BiLSTM to incorporate both character- and word-level embeddings. However, they did not adopt CRF for label prediction. Ma and Hovy, 2016 proposed a truly end-to-end sequence labeling model, using CNN and BiLSTM for character- and word-level presentation and CRF for label prediction. Lample *et al.*, 2016 also proposed a similar architecture, but using RNN for incorporating character-level information. Our single-task neural network architecture comes from Liu *et al.*, 2017a's most recent work, using one BiLSTM layer for character-level representation, stacked with another BiLSTM layer for word-level representation and finally a CRF layer for label prediction. This model takes the whole sentence, instead of single words, as input into the character-level BiLSTM.

In the biomedical domain, state-of-the-art systems often use statistical sequence labeling methods such as CRF with handcrafted features. Ensemble methods, such as combining two CRF models in the *tmChem* system [Leaman *et al.*, 2015], show a further boost of performance. Recent studies tried to apply neural network models for automatic feature generation in BioNER. For example, Habibi *et al.*, 2017 adopted Lample *et al.*, 2016's model and performed analysis on a total of 24 different BioNER datasets. However, these proposed neural network models cannot outperform state-of-the-art systems that utilize handcrafted features for BioNER.

Using multi-task learning in deep neural networks for natural language processing is also attracting great attention in recent years. In the general domain, Collobert and Weston, 2008 applied the multi-task learning framework to some core NLP tasks such as POS-tagging, NP-chunking, NER and semantic role labeling. Søgaard and Goldberg, 2016 also suggested that using a hierarchical neural architecture by putting different tasks at different layers will lead to even better performance. One major challenge for BioNER is the limited available training data for each biomedical entity type. Similar multi-task learning framework can be adopted, but with a difference of modeling different tasks in parallel instead of in a hierarchical order.

Crichton *et al.*, 2017 explored incorporating multi-task learning with CNN for BioNER. In their model, the lookup table and convolutional layers are shared for all tasks. Their results demonstrated the performance boost of multi-task learning compared with the single-task models. However, their best performing model still cannot outperform state-of-the-art systems that utilize handcrafted features for BioNER.

7 Conclusion

We propose a neural network based multi-task learning framework for biomedical named entity recognition. The proposed framework frees experts from manual feature generation and achieves better performance than state-of-the-art systems, even when only limited amount of training data is available. We compare the multi-task neural network model with state-of-the-art systems and other neural network models on five datasets and major biomedical entity types including genes, proteins, chemicals, diseases and cell lines. Results show that the multi-task neural network model achieves significantly better performance than existing models. Furthermore, the analysis suggests that the performance improvement mainly comes from sharing character- and word-level information between different biomedical entity types, resulting in more recognized entities.

There are several further directions with the multi-task model for BioNER. First, combining single-task and multi-task models is a useful direction. Adapting multi-task model to a multi-class classification model would also be useful. Finally, this work suggests that by resolving entity boundary problem and entity type conflicts, we could build a unified system for recognizing multiple types of biomedical entities with high performance and efficiency.

References

Aerts, S., Lambrechts, D., Maity, S., Van Loo, P., Coessens, B., De Smet, F., Tranchevent, L.-C., De Moor, B., Marynen, P., Hassan, B., *et al.* (2006). Gene prioritization through genomic data fusion. *Nature biotechnology*, **24**(5), 537–544.

Al-Aamri, A., Taha, K., Al-Hammadi, Y., Maalouf, M., and Homouz, D. (2017). Constructing genetic networks using biomedical literature and rare event classification. *Scientific reports*, **7**(1), 15784.

Ando, R. K. (2007). Biocreative ii gene mention tagging system at ibm watson. In *Proceedings of the Second BioCreative Challenge Evaluation Workshop*, volume 23, pages 101–103.

Chiu, J. P. and Nichols, E. (2016). Named entity recognition with bidirectional lstm-cnns. *Transactions of the Association for Computational Linguistics*, **4**, 357–370.

Cokol, M., Iossifov, I., Weinreb, C., and Rzhetsky, A. (2005). Emergent behavior of growing knowledge about molecular interactions. *Nature biotechnology*, **23**(10), 1243–1247.

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning (ICML)*, pages 160–167.

Crichton, G., Pyysalo, S., Chiu, B., and Korhonen, A. (2017). A neural network multi-task learning approach to biomedical named entity recognition. *BMC Bioinformatics*, **18**(1), 368.

Dai, H.-J., Chang, Y.-C., Tsai, R. T.-H., and Hsu, W.-L. (2010). New challenges for biological text-mining in the next decade. *Journal of Computer Science and Technology*, **25**(1), 169–179.

Davis, A. P., Grondin, C. J., Johnson, R. J., Sciaky, D., King, B. L., McMorran, R., Wiegers, J., Wiegers, T. C., and Mattingly, C. J. (2017). The comparative toxicogenomics database: update 2017. *Nucleic acids research*, **45**(D1), D972–D978.

Deng, L., Hinton, G., and Kingsbury, B. (2013). New types of deep neural network learning for speech recognition and related applications: An overview. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8599–8603.

Fehrmann, R. S., Karjalainen, J. M., Krajewska, M., Westra, H.-J., Maloney, D., Simeonov, A., Pers, T. H., Hirschhorn, J. N., Jansen, R. C., Schultes, E. A., *et al.* (2015). Gene expression analysis identifies global gene dosage sensitivity in cancer. *Nature genetics*, **47**(2), 115–125.

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448.

Gonzalez, G. H., Tahsin, T., Goodale, B. C., Greene, A. C., and Greene, C. S. (2015). Recent advances and emerging applications in text and data mining for biomedical discovery. *Briefings in Bioinformatics*, **17**(1), 33–42.

Habibi, M., Weber, L., Neves, M., Wiegandt, D. L., and Leser, U. (2017). Deep learning with word embeddings improves biomedical named entity recognition. *Bioinformatics*, **33**(14), i37–i48.

Huang, C.-C. and Lu, Z. (2015). Community challenges in biomedical text mining over 10 years: success, failure and the future. *Briefings in bioinformatics*, **17**(1), 132–144.

Huang, Z., Xu, W., and Yu, K. (2015). Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Jensen, L. J., Saric, J., and Bork, P. (2006). Literature mining for the biologist: from information retrieval to biological discovery. *Nature Reviews Genetics*, **7**(2), 119–129.

Lafferty, J., McCallum, A., and Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the 17th International Conference on Machine Learning (ICML)*, pages 282–289.

Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., and Dyer, C. (2016). Neural architectures for named entity recognition. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 260–270.

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

Leaman, R., Wei, C.-H., and Lu, Z. (2015). tmchem: a high performance approach for chemical named entity recognition and normalization. *Journal of Cheminformatics*, **7**(1), S3.

Leser, U. and Hakenberg, J. (2005). What makes a gene name? named entity recognition in the biomedical literature. *Briefings in bioinformatics*, **6**(4), 357–369.

Li, G., Ross, K. E., Arighi, C. N., Peng, Y., Wu, C. H., and Vijay-Shanker, K. (2015). mirtex: a text mining system for mirna-gene relation extraction. *PLoS computational biology*, **11**(9), e1004391.

Liu, H., Irwanto, A., Fu, X., Yu, G., Yu, Y., Sun, Y., Wang, C., Wang, Z., Okada, Y., Low, H., *et al.* (2015). Discovery of six new susceptibility loci and analysis of pleiotropic effects in leprosy. *Nature genetics*, **47**(3), 267–271.

Liu, L., Shang, J., Xu, F., Ren, X., Gui, H., Peng, J., and Han, J. (2017a). Empower sequence labeling with task-aware neural language model. *arXiv preprint arXiv:1709.04109*.

Liu, P., Qiu, X., and Huang, X. (2017b). Adversarial multi-task learning for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1–10.

Ma, X. and Hovy, E. (2016). End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1064–1074.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3111–3119.

Morris, A. P., Voight, B. F., Teslovich, T. M., Ferreira, T., Segre, A. V., Steinthorsdottir, V., Strawbridge, R. J., Khan, H., Grallert, H., Mahajan, A., *et al.* (2012). Large-scale association analysis provides insights into the genetic architecture and pathophysiology of type 2 diabetes. *Nature genetics*, **44**(9), 981.

Pyysalo, S., Ginter, F., Moen, H., Salakoski, T., and Ananiadou, S. (2013). Distributional semantics resources for biomedical text processing. In *Proceedings of Languages in Biology and Medicine*.

Ramsundar, B., Kearnes, S., Riley, P., Webster, D., Konerding, D., and Pande, V. (2015). Massively multitask networks for drug discovery. *arXiv preprint arXiv:1502.02072*.

Rastegar-Mojarad, M., Ye, Z., Kolesar, J. M., Hebbring, S. J., and Lin, S. M. (2015). Opportunities for drug repositioning from phenome-wide association studies. *Nature biotechnology*, **33**(4), 342–345.

Rebholz-Schuhmann, D., Oellrich, A., and Hoehndorf, R. (2012). Text-mining solutions for biomedical research: enabling integrative biology. *Nature Reviews Genetics*, **13**(12), 829–839.

Smith, L., Tanabe, L. K., nee Ando, R. J., Kuo, C.-J., Chung, I.-F., Hsu, C.-N., Lin, Y.-S., Klinger, R., Friedrich, C. M., Ganchev, K., *et al.* (2008). Overview of biocreative ii gene mention recognition. *Genome Biology*, **9**(S2), S2.

Søgaard, A. and Goldberg, Y. (2016). Deep multi-task learning with low level tasks supervised at lower layers. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 231–235.

Sondhi, P. (2008). A survey on named entity extraction in the biomedical domain.

Szklarczyk, D., Santos, A., von Mering, C., Jensen, L. J., Bork, P., and Kuhn, M. (2015). Stitch 5: augmenting protein–chemical interaction networks with tissue and affinity data. *Nucleic acids research*, **44**(D1), D380–D384.

Szklarczyk, D., Morris, J. H., Cook, H., Kuhn, M., Wyder, S., Simonovic, M., Santos, A., Doncheva, N. T., Roth, A., Bork, P., *et al.* (2017). The string database in 2017: quality-controlled protein–protein association networks, made broadly accessible. *Nucleic acids research*, **45**(D1), D362–D368.

Wang, Z.-Y. and Zhang, H.-Y. (2013). Rational drug repositioning by medical genetics. *Nature biotechnology*, **31**(12), 1080–1082.

Wei, C.-H., Kao, H.-Y., and Lu, Z. (2013). Pubtator: a web-based text mining tool for assisting biocuration. *Nucleic acids research*, **41**(W1), W518–W522.

Willer, C. J., Schmidt, E. M., Sengupta, S., Peloso, G. M., Gustafsson, S., Kanoni, S., Ganna, A., Chen, J., Buchkovich, M. L., Mora, S., *et al.* (2013). Discovery and refinement of loci associated with lipid levels. *Nature genetics*, **45**(11), 1274.

Xie, B., Ding, Q., Han, H., and Wu, D. (2013). mircancer: a microrna–cancer association database constructed by text mining on literature. *Bioinformatics*, **29**(5), 638–644.

Zhou, G. and Su, J. (2004). Exploring deep knowledge resources in biomedical name recognition. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, pages 96–99.

Zhou, X., Menche, J., Barabási, A.-L., and Sharma, A. (2014). Human symptoms–disease network. *Nature communications*, **5**, 4212.