Journal of Biomedical Informatics 112 (2020) 103609

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Original Research

Character level and word level embedding with bidirectional LSTM – Dynamic recurrent neural network for biomedical named entity recognition from literature

Sudhakaran Gajendran [a,*], Manjula D [a], Vijayan Sugumaran [b,c]

[a] Department of Computer Science and Engineering, College of Engineering Guindy, Anna University, Chennai, India
[b] Center for Data Science and Big Data Analytics, Oakland University, Rochester, MI, USA
[c] Department of Decision and Information Sciences, School of Business Administration, Oakland University, Rochester, MI, USA

ARTICLE INFO

Keywords:
Biomedical named entity recognition
Embeddings
Deep neural networks
Bidirectional LSTM
Dynamic RNN
CRF

ABSTRACT

Named Entity Recognition is the process of identifying different entities in a given context. Biomedical Named Entity Recognition (BNER) is the task of extracting chemical names from biomedical texts to support biomedical and translational research. The aim of the system is to extract useful chemical names from biomedical literature text without a lot of handcrafted engineering features. This approach introduces a novel neural network architecture with the composition of bidirectional long short-term memory (BLSTM), dynamic recurrent neural network (RNN) and conditional random field (CRF) that uses character level and word level embedding as the only features to identify the chemical entities. Using this approach we have achieved the F1 score of 89.98 on BioCreAtIvE II GM corpus and 90.84 on NCBI corpus by outperforming the existing systems. Our system is based on the deep neural architecture that uses both character and word level embedding which captures the morphological and orthographic information eliminating the need for handcrafted engineering features. The proposed system outperforms the existing systems without a lot of handcrafted engineering features. The embedding concept along with the bidirectional LSTM network proved to be an effective method to identify most of the chemical entities.

1. Background

With the dramatic improvements in the field of bioinformatics, extracting information from text and analyzing the association between the chemical entities has received more attention in the past few years [1]. Entity Recognition (ER) meant to extract and recognize the entities from any text. Biomedical Named Entity Recognition (BNER) is the process of extracting chemical entities from text [2]. BNER gets more and more attention from the researchers since it is a fundamental task in biomedical information extraction [3].

Several machine learning techniques have been used over the past few years to extract chemical entities from literature [4,5]. Machine learning algorithms use different kinds of feature sets which include both syntactic features as well as domain specific features based on the domain. It produces promising results with the help of several executable tools [6] and also due to the availability of annotated datasets [7,8]. However, the result of the machine learning techniques fully depends on the selection of the feature set. The process of selecting the appropriate features is not an easy task since it needs more domain knowledge of how a specific feature will work in a machine learning algorithm [9]. Also, machine learning techniques are not suitable to identify the long-term dependencies between the chemical entities from papers in the literature.

The recent advances in the field of deep learning have encouraged researchers to overcome the issues that arise from machine learning algorithms. Deep learning, a subset of machine learning utilizes a hierarchical level of artificial neural networks to carry out the process of machine learning together with the concept of learning by example [10,11]. Most deep learning methods use neural network architecture which is why they are often referred to as deep neural networks (DNN) [12]. DNN have the capability of selecting the features automatically from the vector values [13,14]. Also, Bidirectional Long Short Term Dependencies (BLSTM) which is a special kind of Recurrent Neural Network (RNN) is used to identify the long-term dependencies between

* Corresponding author.
E-mail addresses: gsudhaak@gmail.com, sudhakar@cs.annauniv.edu (S. Gajendran), dmanju62@gmail.com (M. D), sugumara@oakland.edu (V. Sugumaran).

https://doi.org/10.1016/j.jbi.2020.103609
Received 18 March 2020; Received in revised form 14 October 2020; Accepted 22 October 2020
Available online 26 October 2020

the chemical entities [10,15,16]. Advances in the field of embedding, i.e., in both character level and word level embedding have benefitted the researchers in three ways 1) words can be converted and denoted as vector values, 2) can be used in unlabeled corpus, and 3) can work efficiently with long words [17–19].

In this paper, we have proposed a novel neural network architecture with the composition of bidirectional long short-term memory (BLSTM), dynamic recurrent neural network (DRNN) and conditional random field (CRF) that uses character level embedding and word level embedding as the only features to identify chemical entities. We have incorporated a three level BLSTM-DRNN network to capture character and word representation followed by the LSTM – CRF model [20].

The contributions of the proposed systems are:

- A three level BLSTM-DRNN model to capture the character and word representation.
- Dynamic RNN to reduce the computation time and space.
- Comparison of different variants of the proposed system to study the impact and effectiveness of word embedding.
- Comparison on three publicly available datasets with the benchmarking methods to illustrate the effectiveness of the proposed system.

We have evaluated our model using the BioCreAtIvE II GM corpus [21], NCBI corpus and JNLPBA corpus. Experimental results show that our BLSTM-RNN model with the embeddings achieves better performance compared to the existing state-of-the-art systems.

The remainder of this article is organized as follows. Section 2 describes the related works in BNER. Section 3 introduces our BLSTM-DRNN model for biomedical named entity recognition. Section 4 describes the experimental setup. Results and comparison with the existing systems are described in Section 5. Finally, conclusions and future work is discussed in Section 6.

2. Related works

The applications of text mining in the field of bioinformatics combines biology with computer science and addresses the problem of collection, processing and analyzing data related to healthcare [22]. The most useful task in biomedical text mining is the recognition of useful information residing in a large number of papers in the literature [2]. Generally, the prior biomedical NER systems have utilized three categories of methods: Rule-based methods, dictionary-based methods and machine learning methods [8,23]. The machine learning methods are more powerful among the three and have been tried in almost all possible aspects [7,24]. Most of the machine learning approaches produce better results than the other methods, however, the results mostly depends on the selection of the feature sets [25].

Rule-based methods depend on the static rules which are framed to identify the chemical entities [2]. The disadvantage of this approach is that rules cannot keep pace with the developments and emerging initiatives [23]. The rule-basedapproach simply cannot cover every circumstance or eventuality, leaving gaps that could potentially be exploited [22,26]. The dictionary-based approaches rely on the predefined dictionaries or gazetteers which contain all possible lists of biomedical texts [22,26]. Lemmas are used to recognize the most identical entity from the dictionary [23].

Most existing Named Entity Recognition methods are based on machine learning algorithms. The widely used machine learning algorithms include Hidden Markov Models (HMM) [27], Maximum Entropy Markov Models (MEMM) [28], SVM [29], SSVM [30] and CRF [31,32] which are used to extract semantic and syntactic features from the manually annotated datasets and identify the entities from the research texts. Feature-based systems concentrate on identifying potentially discriminatory characteristics to reflect the characteristics of the data. In addition to the typical bag-of -word features, studies exploit the advantages of different attributes including context [32], and a mixture of lexical and domain based features [4]. Also diverse attributes comprising of lexical, syntactic and negation based features obtained from parse trees [25] are also used. Nonetheless, for training supervised classifiers, these methods usually depends on manual handcrafted features to produce selective features. In general, machine learning techniques produced promising results due to the availability of annotated datasets. However, the results of machine learning techniques fully depends on the selection of the feature set [9]. The process of selecting the appropriate features is not an easy task since it needs more domain knowledge of how a specific feature will work in the machine learning algorithm.

Deep learning techniques have been greatly explored in recent years for the automated learning of features for BNER and association extraction even from the scrambled form of data [12,33]. Named Entity Recognition systems can be trained without expensive manual feature engineering using deep neural networks [14]. Gridach [18] proposed a composition of BiLSTM-CRF model to identify the entities. Gridach used a character embedding using BiLSTM model followed by pre-trained word representation. Cho et al. [34] incorporated a two layer system comprising of CNN and BiLSTM for character embedding along with pre-trained word representation to feed into fully connected BiLSTM-CRF model. However, these models incorporated pre-trained word representation to improve the efficiency of the system.

Recent studies have explored the possibility of incorporating multi-task learning, which take advantage of training a single model to work on different datasets [35]. Yoon et al. [36] proposed CollaboNet with multiple layers of BiLSTM-CRF model along with a pre-trained word representation from multiple sources including wikipedia and PudMed. Lee et al. [37] incorporated a BioBERT: pre-trained word representation system trained from a massive unlabeled datasets. These models have produced promising results by capturing representation from different domains [37]. However, these models depend on the pre-trained word representation and also training these models have taken enormous amount of time. In our work, we have utilized a Bidirectional LSTM - Dynamic RNN based method to incorporate character and a novel word representation to capture the morphological and orthographic information without pre-trained representation.

We have proposed a novel neural network architecture with the composition of bidirectional long short-term memory (BLSTM), dynamic recurrent neural network (DRNN) and conditional random field (CRF) that uses character level embedding and word level embedding as the only features to identify chemical entities. In addition to that, DRNN dynamically computes and allocates the sequence length for each batch of sequence. The DRNN significantly reduces the computation time and space even if the batch sizes are unequal. Later, the values are combined and fed into BLSTM-DRNN model to capture the context information for each word. Then on top of the DNN, a CRF layer is used to decode the vector and to identify the entities for the whole sentence. We have evaluated our model using the BioCreAtIvE II GM corpus [21], NCBI corpus and JNLPBA corpus and compared it with the state-of-the-art systems [18,38,39]. Our proposed approach is described in detail in the following section.

3. Proposed system

The proposed system introduces a novel neural network architecture with the composition of bidirectional long short-term memory (BLSTM), dynamic recurrent neural network (DRNN) and conditional random field (CRF) that uses character level embedding and word level embedding as the only features to identify the chemical entities. The biomedical Named Entity Recognition system consists of various processes like context dependent tokenization, segmentation, Dictionary creation, LSTM model generation.

The architecture of the proposed system is shown in Fig. 1. In this work, after tokenization using Context Dependent Tokenizer (CDT),

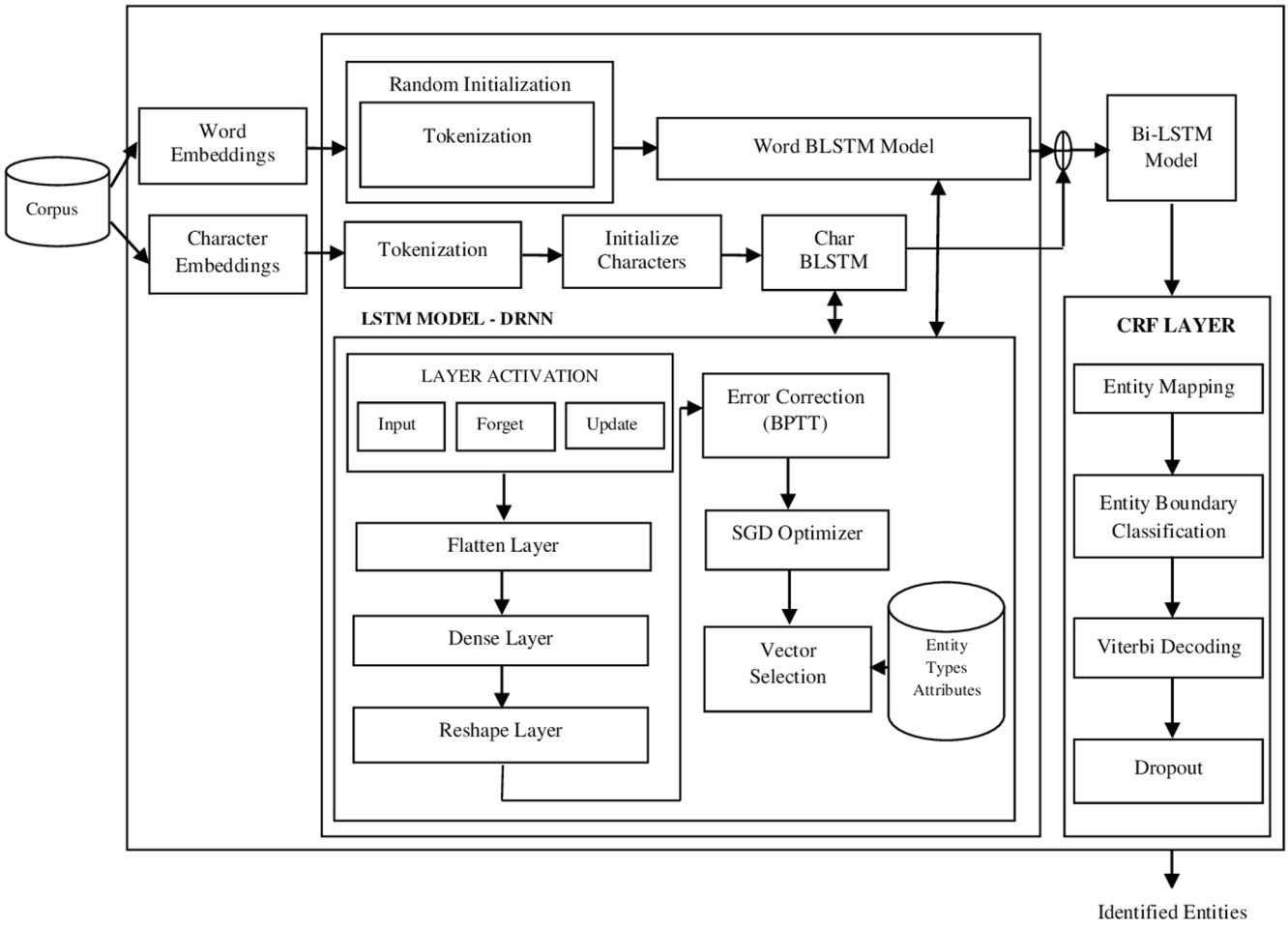

Fig. 1. Overall Architecture of the Proposed System.

Character embedding and word embedding converts each character and word into their corresponding vector values using separate BLSTM-DRNN model and the output from both the embeddings are concatenated and fed into BLSTM-DRNN models which passes through a number of layers such as flatten, dense and reshape layer. These layers are used to refine and finalize the output vector values for each input. Later, the Backpropagation through Time (BPTT) and Stochastic Gradient Descent (SGD) optimizers are used to identify and minimize the error in the BLSTM model. The former is used to backpropagate the vector values again to the layers if the input is wrongly identified whereas the latter is used to minimize the error in the vector values. Then on the top of the DNN, a CRF layer is used to decode the vector and to identify the entities for the whole sentence [12].

The BLSTM model reads the vector values from the embedding in both directions i.e., in both forward and backward direction which cross verifies the vector values against each other to make sure that the correct vector values are being read [40]. The system also uses Dynamic RNN that dynamically computes and allocates the sequence length for each batch of sequence. The DRNN significantly reduces the computation time and space even if the batch sizes are unequal. Thus, the BLSTM-DRNN model ensures that the output vector values are generated without any error and also minimizes the amount of processing time considerably.

3.1. Tokenization

The Context Dependent Tokenizer (CDT) tokenizes only the biomedical terms and assigns unique tokens to them. It segments the words and tokenizes each word and puts them into the separate columns. By using character representation the unique index value for a word in each sentence is generated. For each character in memory cell, a unique index value is generated. The CDT algorithm gets the text from the literature as the input and assigns token for each word. CDT algorithm processes each word in the sentence and assigns unique token for those words.

Algorithm 1 describes the process of tokenization where LT contains list of words in the vocabulary and LW contains the actual input. If the input word LW is not available in LT then that word is split into subwords and tokens are assigned to them with a special symbol. This process retains the contextual meaning and avoids tokenizing OOV words to NULL.

Algorithm.1 CDT(X)

```
1) X = Input file
2) Read X
3) LT[tags]list of tags      // LT contains list of words in vocabulary
4) LW[tags]list of words     // LW contains the actual input list
5) for every word in LW
6)    if tags is not equal to zero
7)       if LW are unique and available in LT
8)          Assign token to the words;
9)       elseif it is not available in LT
10)         Split the word into subwords and assign tokens with special symbol
11)      else
12)      set the type property of token = NULL
13) return token
```

3.2. Character and word embeddings

We have proposed a novel word embedding approach that initializes the random vector values for each word in the first layer and then updates the values in each layer eliminating the use of pre-trained word representation. After tokenization, tokens are assigned as the input to char BLSTM and word BLSTM model. For both character and word embedding, the tokenized input is fed into BLSTM-DRNN model. The bidirectional LSTM model fits the input word with initial random vectors into both forward and backward LSTM. The cell from both LSTM is carried over to Dynamic RNN which calculates the sequence length dynamically for each input batch. So, the DRNN processes the vectors through the hidden units only upto the maximum sequence length for each batch and updates the vector values. An embedding is a mapping from discrete objects, such as words, to vectors of real numbers. The individual dimensions in these vectors typically have no inherent meaning; here all values contribute to defining an object [17].

Classifiers and neural networks more generally, work on vectors of real numbers. Embeddings maps objects to vectors. Word and character embedding is the idea of taking them individually from the context and use them in training to gain their original identity vector value for the deep learning process [41]. It involves building a low-dimensional vector representation from the corpus of text, which preserves the contextual similarity of words [18]. In character level embedding, a vector value for a single word is constructed from the character n grams. Since same n grams characters are shared across words, every single word's vectors can be formed even for out-of-vocabulary (OOV) words, new and infrequent words. Fig. 2 shows the char BLSTM process where a single word CD28 generates character n grams and a vector for that word is constructed. Since the vector values are calculated using n grams for each character, character embedding also handles misspelled words. For example, vector value for the drug name "zithromax" is not same as the vector for the word with typo "zithormax" because the two vectors from the n grams are very different.

Similarly, Fig. 3 shows the word level embedding process where the sentence "Activation of the CD28 surface receptor" generates different word n grams and a vector representation for each word is constructed. Word BLSTM accepts a random vector as the initial value for each word and those vectors pass through the layers of BLSTM model in both directions. Upon completion of the layers, the vector values are updated each time. Word embedding is used to capture the entity relation with the other entity, semantic similarity and context of the entity in the document [42,43].

3.3. Bi-Directional LSTM - DRNN model

The vector values obtained after the concatenation of character and word representation is passed to the Bidirectional LSTM (BLSTM) cell [44]. The system uses a dynamic RNN that dynamically computes and allocates the sequence length for each batch of sequence. The DRNN significantly reduces the computation time and space even if the batch sizes are unequal. The BLSTM model contains three layer activation functions: Input, Forget and Update layer [45–47]. The following formulas are used to update the layers of BLSTM at each time t.

$$f_t = \sigma(w_f[h_{t-1}, x_t] + b_i) \quad (1)$$

$$i_t = \sigma(w_t[h_{t-1}, x_t] + b_i) \quad (2)$$

$$\widehat{c}_t = tanh(w_t[h_{t-1}, x_t] + b_i) \quad (3)$$

$$c_t = f_t{*}c_{t-1} + i_t{*}\widehat{c}_t \quad (4)$$

where w_f, w_t are the weight matrices for hidden state h_t and x_t is the input vector and b_i represents the bias vector. σ and tanh represents the sigmoid and hyperbolic tangent function respectively. Here i_t and$\widehat{c}_t$represents the input layer function, f_t represents the forget layer function, and c_t represents the update layer function [18].

The BLSTM - DRNN Model has the core layers such as flatten, reshape and dense layers that do the process of refining the embedding values [8]. The flatten and dense layers rounds off the vector values and the reshape layer compares the round off value with the predefined archived values to finalize the output vector. If the vector value is much deviated from the archived value, then it is back propagated to determine the correct vector value [47].

The bidirectional dynamic RNN model with Long Short-Term Memory uses purpose-built memory cells to store information and is better at finding and exploiting long range data dependencies [48,49]. Using those vector values it processes in both directions with the automatically calculated sequence length for each batch. Dynamic RNN makes sure that the system works well even with the different sizes of sequence texts. Thus, dynamic RNN improves the speed and efficiency of the system. SGD optimizer is used to find the minimum value and to

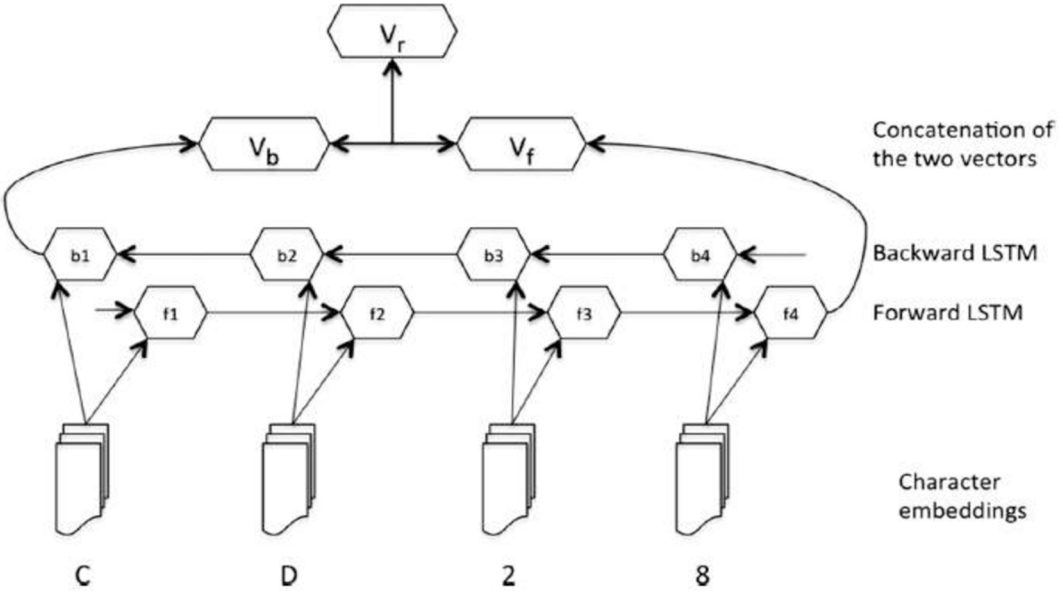

Fig. 2. Character Embedding of BLSTM.

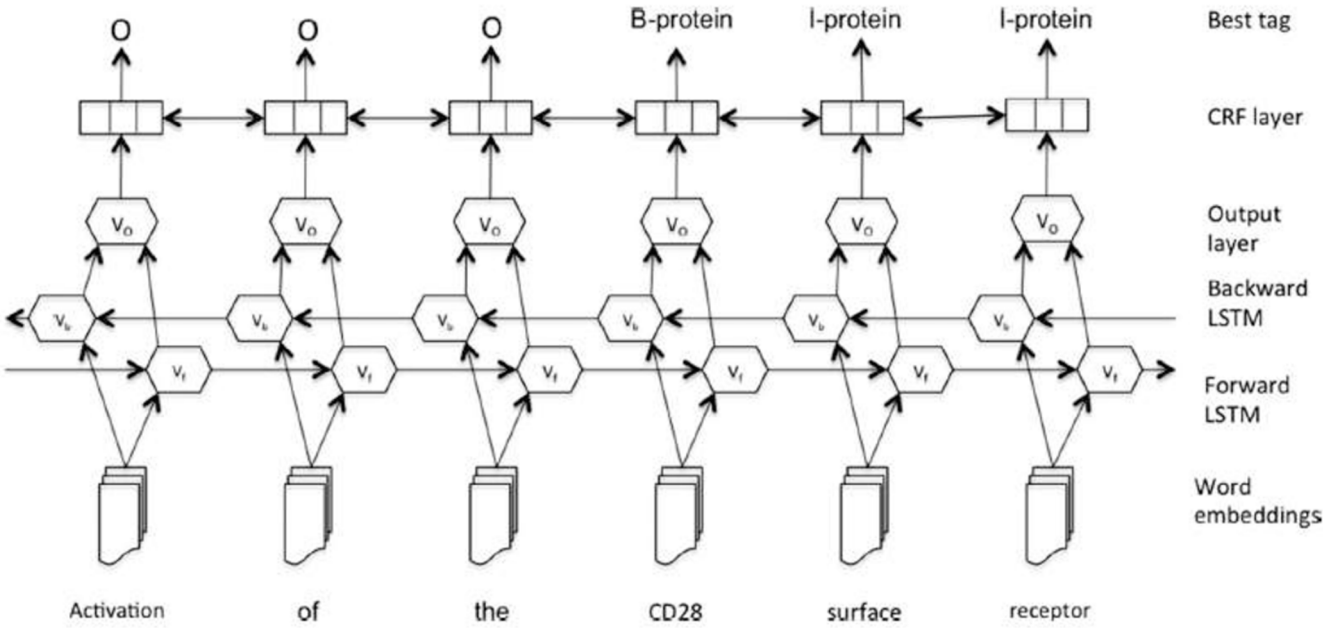

Fig. 3. Word Embedding of BLSTM.

reduce redundancy of same vectors.

The algorithm for the proposed bidirectional LSTM - dynamic RNN network is given below in Algorithm 2:

Algorithm.2 BLSTM-DRNN

1) **input** = Input sequence
2) cell **fw** = Forward LSTM cell
3) cell **bw** = Backward LSTM cell
4) initial state **fw** = zero
5) initial state **bw** = zero
6) **seq len** = calculate maximum sequence length for each batch of **fw** and **bw**
7) **output fw**, **output state fw** = dynamic rnn(input,seq len,seq dim,batch dim)
8) reverse()
9) *if* length(**input**) not none
10) return reversed sequence
11) *else*
12) mark it as end of sentence
13) with variable scope as **bw** scope
14) **input reverse** = reverse(**input**)
15) **tmp, outptut state bw** = dynamic RNN (input, seq len,seq dim,batch dim)
16) **output bw** = reverse (**tmp**)
17) **outputs** = concatenate outputs of forward and backward cells
18) **outputs states** = concatenate outputs of forward and backward cells

3.4. BackPropagation through time (BPTT)

The BLSTM networks are trained using the BackPropagation Through Time (BPTT). In a recurrent neural network, errors can be propagated deep into many layers, i.e. more than 2 layers to capture longer history information [50,51]. This is referred to as unfolding. In an unfolded RNN, the recurrent weight is duplicated spatially for a random number of time steps. That is, the reshape layer of BLSTM network compares the computed vector value with the archived vector value and if any deviation occurs between the values, then the computed value is unfolded recursively to find the error. After the error is identified, weights are folded up to one big change for each unfolded weights. The algorithm for BPTT is given below.

Algorithm.3 BackPropagation_Through_Time(a, y):

Training of the model by unfolding the network and identifying the error rate and updating the values.
// a[t] is the input at time t. y[t] is the output

(*continued on next column*)

(*continued*)

Algorithm.3 BackPropagation_Through_Time(a, y):

1) unfold the network to contain **k** instances of **f**
2) *do* until stopping criteria is met:
3) **x** = the zero-magnitude vector; *// x is the current context*
4) *for* **t** from **0 to n - k** *// t is time. n is the length of the training sequence*
5) set the network inputs to **x, a [t] , a [t + 1] ,** …**, a [t + k-1]**
6) **p** = forward-propagate the inputs over the whole unfolded network
7) **e = y [t + k] - p;** *// error = target - prediction*
8) back-Propagate the error, **e**, back across the whole unfolded network
9) sum the weight changes in the **k** instances of **f** together.
10) update all the weights in **f** and **g**.
11) **x = f(x, a [t]);** *// compute the context for the next time-step*
12) *end*
13) *end*

3.5. Stochastic Gradient Descent (SGD)

The Stochastic Gradient Descent (SGD) [52] algorithm updates weights in each run to converge quickly by choosing the random subsamples called mini-batch from the training data. Since a small selection of subsamples is used, it computes errors and updates weights in faster iterations. Also, it often helps to move towards convergence more quickly because the gradient is averaged at each step over more training examples.

Algorithm.4 SGD model

1) *while* not converged *do*
2) randomly shuffle examples in training set
3) *for* **i = 1, …,N** *do*
4) $W = W - \eta \nabla Q_i(W)$ //where $\eta$is the learning rate
5) *end*

3.6. CRF layer

The model consists of entity mapping and entity boundary classification. In the entity mapping phase the resultant vectors are grouped into corresponding labels. In the second phase, Viterbi method [44] is used for refinement of the output. This will identify the tags by considering its neighbors. The CRF layer is added at the last to group the entity based on their pattern recognized by comparing with the

neighbors [20,53] after dropout process.

4. Experimental setup

In this section, we describe the corpus statistics, hyper parameter settings for training the network and the evaluation metrics used to evaluate the proposed system. Our proposed system is implemented in python using an open source network library called Keras. Keras is capable of running on top of Theano [54], Tensorflow and Deeplearning4j with a focus on enabling fast experimentation. In our system, Keras is used on the top of Tensorflow which is an open source software library for dataflow programming across a range of tasks. Tensorflow is a symbolic math library, and also used for machine learning applications such as neural networks.

4.1. Datasets and hyper parameters

We evaluated our proposed system with three different benchmark datasets: BioCreAtIvE II GM corpus [21], JNLPBA corpus [38] and NCBI corpus [39]. The BNER corpus statistics are listed in Table.1. The JNLPBA corpus consists of 18,546 sentences for training and 3856 sentences for testing purpose. Also the training sentences are split in the ratio of 9:1 for development purpose. Similarly, the NCBI corpus consists of 5145, 787 and 960 sets for training, development and testing purpose respectively.

The dimensions of character level and word level embedding were 100 and 200 respectively. We used 300 hidden units for BLSTM model with a dropout of 0.5 for all the three BLSTM units (char BLSTM, word BLSTM and last hidden unit of BLSTM). The SGD optimizer in our experiment used the learning rate parameter of 0.01 with the decay of 0.05 for each epoch. As we use dynamic RNN, the sequence length changes dynamically according to the maximum sequence length in every mini batch size of 128.

4.2. Evaluation metrics

The performance of the entire system is evaluated using the standard parameters Precision (P), Recall (R), and F1 score (F1) [55,56].

$$P = \frac{TP}{TP + FP} \tag{5}$$

$$R = \frac{TP}{TP + FN} \tag{6}$$

$$F1 = \frac{2*P*R}{P + R} \tag{7}$$

where TP is the number of correct entity returned by the system, FP is the number of incorrect entities returned by the system and FN is the number of missing entities. In the context of biomedical NER, precision is the percentage of biomedical named entities identified by the system that is correct and recall is the percentage of relevant entities that have been retrieved over the total amount of relevant entities. F1 score (also F-score or F-measure) is a measure of a test's accuracy. It considers both the precision (P) and the recall (R) of the test to compute the score.

Table 1
Biomedical NER Corpus Statistics.

Corpus	Size	Type
BioCreAtIvE II GM corpus	20,000 Sentences	Gene/Proteins
JNLPBA corpus	2404 Abstracts	Gene/Proteins
NCBI corpus	793 Abstracts	Disease

5. Results and discussions

5.1. Comparison with variants of proposed system

The evaluation of the proposed system was experimented in BioCreAtIvE II GM corpus. Then the top attained model is used to run on JNLPBA and NCBI corpus. Initially the evaluation of the proposed system is carried out on four different variants: LSTM with word embedding, Bidirectional LSTM with word embedding, Bidirectional LSTM with character embedding, and finally the Bidirectional LSTM with character and word level embedding as well as CRF. The baseline models are finalized in such a way to exploit the performance of the word embedding. Table 2 shows the results and it can be seen that last system produces the best result with 89.98 as the F1 score. Using the word level embedding with dropout improves system by the F1 score of 3.43 points and proves the importance of word level embedding in named entity recognition [17]. Also using CRF as the top layer in the architecture enhances the performance of the system. Using dynamic RNN provides the advantage of dynamically calculating the sequence length which significantly reduces the overall running time.

As shown in Table.2, the second model used Bidirectional LSTM with word embeddings (BLSTM+WE) and achieved 83.85 points for F1-score. The third model uses Bidirectional LSTM with characterembedding (BLSTM+CE) and produced the F1 score of 86.55 points which is better than the previous model. The last outperforms all the other models by attaining the top F1 score of 89.98, which is an increase of 3.43 points in F1 score for the BioCreAtIvE II GM corpus. The last model is used to evaluate on JNLPBA and NCBI corpus.

5.2. Comparison with the existing systems

In this section we compare our proposed system with the existing systems using three different corpuses: BioCreAtIvE II GM corpus, NCBI corpus and JNLPBA corpus. Table3 shows the comparison between our proposed system and other existing systems. First, we compared our system with the standard state-of-the-art systems for each corpus reported in the recent literature. Specifically, we have compared the results from our system with those from the state-of-the-art systems as follows: results from Sachan et al. (2018) [39] for the BioCreAtIvE II GM corpus, results from Xu et al. (2019) [38] for the NCBI corpus, and results from Gridach (2017) [18] for the JNLPBA corpus. Next, we also compared our work with other systems reported in the most recent literature such as Wang et al. [35], Yoon et al. [36], Lee et al. [37] and Cho et al. [34].

Lee et al. [37] achieved second best results using BioBERT (Bidirectional Encoder Representations from Transformers for Biomedical Text Mining) in all the three corpuses. BioBERT model outperformed the other systems with a domain specific representation pre-trained on a massive biomedical corpus. Finally, the character level word embedding based BLSTM-CRF model called CollaboNet by Yoon et al. [36] slightly outperformed our model in the JNLPBA corpus with the F1 score by 1.45% but our model retains the highest precision value of 76.95. Moreover, our proposed system outperformed the existing systems with the highest F1 score of 89.98 in the BioCreAtIvE II GM corpus and 90.84 in the NCBI corpus. Even though our system is outperformed in JNPLBA corpus, we achieved state-of-the-art performance by 5.26% higher in BioCreAtIvE II GM corpus and 1.13% higher in NCBI corpus defending

Table 2
Results on BioCreAtIvE II GM corpus.

Systems	Precision	Recall	F1 Score
LSTM + WE	86.54	76.05	80.96
BLSTM + WE	85.63	82.13	83.85
BLSTM + CE	87.85	85.29	86.55
BLSTM + CE + WE + CRF + dropout	90.46	89.05	89.98

Table 3
Comparison with State-of-art-art Systems.

	BioCreAtIvE II GM			JNLPBA			NCBI		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
State-of-the-art System	81.81	81.57	81.69	74.16	77.66	75.87	88.30	89.00	88.60
Wang et al. (2018) [35] "BLSTM-CRF"	82.10	79.42	80.74	70.91	76.34	73.52	85.86	86.42	86.14
Yoon et al. (2019) [36] "CollaboNet"	80.49	78.99	79.73	74.43	83.22	**78.58**	85.48	87.27	86.36
Lee et al. (2019) [37] "BioBERT"	84.32	85.12	84.72	72.24	**83.56**	77.49	88.22	**91.25**	89.71
Cho et al. (2020) [34] "CNN-BLSTM"	–	–	–	71.89	79.07	73.51	86.75	87.11	86.93
Proposed System	**90.46**	**89.05**	**89.98**	**76.95**	75.84	77.13	**89.31**	90.45	**90.84**

the massively pre-trained multi-task models like BioBERT and CollaboNet.

The higher precision and recall score are attributed to the efficiency of word embedding and Bidirectional LSTM system. Adding word and character level embeddings allowed the system to learn features automatically from the rich biomedical texts instead of hand-engineering them. The proposed system with a basic but technically powerful word embedding and BLSTM-DRNN model shows better performance than massively pre-trained models. Also the dynamic RNN improves the efficiency and reduces the computation time.

5.3. Dropout effect

Before inputting to the bidirectional LSTM we apply dropout technique on the final embedding layer. The results showed that dropout increases the efficiency of the models on all the three datasets. Table.4 compares theoutcomes with and without the use of dropout where all other hyperparameters for the model chosen remain the same. The reason behind this outcome is because of the use of dropout we allow the model to belong to embedding at both word and character level.

6. Conclusion and future works

The proposed system introduces a novel neural network architecture with the composition of bidirectional long short-term memory (BLSTM), dynamic recurrent neural network (DRNN) and conditional random field (CRF) that uses character level embedding and word level embedding as the only features for biomedical named entity recognition. Our approach outperforms the existing systems and achieves state of the art result for BioCreAtIvE II GM corpus and NCBI corpus. Our system is based on the deep neural architecture that uses both character and word level embedding which captures the morphological and orthographic information eliminating the need for handcrafted engineering features. Also, dynamic RNN with the process of automatically identifying the sequence length reduces the computation time and increases the efficiency of the system. We have achieved the F1 score of 89.98 with the BioCreAtIvE II GM corpus and 90.84with the NCBI corpus and outperformed the existing systems. Thus the word embedding concept along with the bidirectional LSTM network proved to be an effective method to identify most of the chemical entities.

As part of our future work, we plan to evaluate our proposed system using more datasets. Also, we plan to use our model with multi-task learning approach of sharing the same parameters across different NER systems with different datasets. We also plan to expand our system and add appropriate mechanisms to serve as a search engine for biomedical entities recognition.

CRediT authorship contribution statement

Sudhakaran Gajendran: Conceptualization, Methodology, Software, Writing - original draft. **D. Manjula:** Visualization, Data curation, Supervision. **Vijayan Sugumaran:** Writing - review & editing, Validation, Investigation.

Table 4
Results with and without dropout on datasets.

	F1-score on JNLPBA corpus	F1-score on BioCreAtIvE II GM corpus	F1 score on NCBI corpus
Without dropout	75.90	87.77	88.62
With dropout	77.13	89.98	90.84

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jbi.2020.103609.

References

[1] W. Zheng, H. Lin, L. Luo, et al., An attention-based effective neural model for drug-drug interactions extraction, BMC Bioinf. 18 (445) (2017), https://doi.org/10.1186/s12859-017-1855-x.
[2] X. Wang, C. Yang, R. Guan, A comparative study for biomedical named entity recognition, Int. J. Mach. Learn. Cyber 9 (3) (2018) 373–382, https://doi.org/10.1007/s13042-015-0426-6.
[3] R. Danger, F. Pla, A. Molina, P. Rosso, Towards a protein–protein interaction information extraction system: recognizing named entities, Knowl.-Based Syst. 57 (2014) 104–118, https://doi.org/10.1016/j.knosys.2013.12.010.
[4] A.S. Al-Hegami, A.M.F. Othman, Bagash FT.A biomedical named entity recognition using machine learning classifiers and rich feature set, Int. J. Comput. Sci. Netw. Secur. 17 (1) (2017) 170–176.
[5] J. Atkinson, V. Bull, A multi-strategy approach to biological named entity recognition, Expert Syst. Appl. 39 (2012) 12968–12974, https://doi.org/10.1016/j.eswa.2012.05.033.
[6] G. Gonzalez, R. Leaman, Banner: an executable survey of advances in biomedical named entity recognition, Pac Symp. Biocomput. 13 (2008) 652–663, https://doi.org/10.1142/9789812776136_0062.
[7] M. Rais, A. Lachkar, A. Lachkar, S. Ouatik, A comparative study of biomedical named entity recognition methods based machine learning approach, in: Third IEEE International Colloquium in Information Science and Technology (CIST); 2014 Oct 20-22; Tetouan, Morocco, 2014, https://doi.org/10.1109/CIST.2014.7016641.
[8] S. Eltyeb, N. Salim, Chemical named entities recognition: a review on approaches and applications, J. Cheminformatics 6 (17) (2014), https://doi.org/10.1186/1758-2946-6-17.
[9] K. Yamamotoa, T. Kudob, A. Konagayac, Y. Matsumoto, Use of morphological analysis in protein name recognition, J. Biomed. Inform. 37 (2004) 471–482, https://doi.org/10.1016/j.jbi.2004.08.001.
[10] F. Tong, Z. Luo, D.A. Zhao, Deep network based integrated model for disease named entity recognition, in: IEEE International Conference on Bioinformatics and Biomedicine (BIBM); 2017 Nov 13-16; Kansa City, MO, USA, 2017, https://doi.org/10.1109/bibm.2017.8217723.
[11] F. Li, M. Zhang, G. Fu, D. Ji, A neural joint model for entity and relation extraction from biomedical text, BMC Bioinf. 18 (198) (2017), https://doi.org/10.1186/s12859-017-1609-9.
[12] F. Li, M. Zhang, B. Tian, B. Chen, G. Fu, D. Ji, Recognizing irregular entities in biomedical text via deep neural networks, Pattern Recogn. Lett. 105 (2018) 105–113, https://doi.org/10.1016/j.patrec.2017.06.009.

[13] L. Yao, H. Liu, Y. Liu, X. Li, M.W. Anwar, Biomedical named entity recognition based on deep neutral network, Int. J. Hybrid Inform. Technol. 8 (8) (2015) 279–288, https://doi.org/10.14257/ijhit.2015.8.8.29.
[14] W. Zheng, H. Lin, Z. Li, et al., An effective neural model extracting document level chemical induced disease relations from biomedical literature, J. Biomed. Inform. 83 (2018) 1–9, https://doi.org/10.1016/j.jbi.2018.05.001.
[15] S.K. Sahu, A. Anand, Drug-drug interaction extraction from biomedical text using long short term memory network, J. Biomed. Inform. 86 (2017) 15–24, https://doi.org/10.1016/j.jbi.2018.08.005.
[16] W. Gunawan, D. Suhartono, F. Purnomo, A. Ongko, Named-entity recognition for Indonesian language using bidirectional LSTM-CNNs, Procedia Comput. Sci. 135 (2018) 425–432, https://doi.org/10.1016/j.procs.2018.08.193.
[17] B. Tang, H. Cao, X. Wang, Q. Chen, H. Xu, Evaluating word representation features in biomedical named entity recognition task, Biomed Res. Int. (2014), https://doi.org/10.1155/2014/240403.
[18] M. Gridach, Character-level neural network for biomedical named entity recognition, Biomed. Informat. 70 (2017) 85–91, https://doi.org/10.1016/j.jbi.2017.05.002.
[19] Y. Wang, S. Liu, N. Afzal, et al., A comparison of word embeddings for the biomedical natural language processing, J. Biomed. Inform. 87 (2018) 12–20, https://doi.org/10.1016/j.jbi.2018.09.008.
[20] L. Li, Y. Jiang, Biomedical named entity recognition based on the two channels and sentence-level reading control conditioned LSTM-CRF, in: IEEE International Conference on Bioinformatics and Biomedicine (BIBM); 2017 Nov 13-16; Kansa City, MO, USA, 2017, https://doi.org/10.1109/bibm.2017.8217679.
[21] A. Yeh, A. Morgan, M. Colosimo, L. Hirschman, BioCreAtIvE Task 1A: gene mention finding evaluation, BMC Bioinf. 6 (S2) (2005), https://doi.org/10.1186/1471-2105-6-S1-S2.
[22] A. Ghoulam, F. Barigou, G. Belalem, F. Meziane, Query expansion using medical information extraction for improving information retrieval in French medical domain, Int. J. Intell. Inf. Technol. 14 (3) (2018) 1–17, https://doi.org/10.4018/IJIIT.2018070101.
[23] S.A. Akhondi, K.M. Hettne, E. Van Der Horst, E.M. Van Mulligen, J.A. Kors, Recognition of chemical entities: combining dictionary-based and grammar-based approaches. Journal of, Cheminformatics 7 (S10) (2015), https://doi.org/10.1186/1758-2946-7-S1-S10.
[24] A. Jain, A. Arora, Named entity system for tweets in Hindi language, Int. J. Intell. Inf. Technol. 14 (4) (2018) 55–76, https://doi.org/10.4018/IJIIT.2018100104.
[25] S.P. Umare, N.A. Deshpande, A survey on machine learning techniques to extract chemical names from text documents, (IJCSIT) Int. J. Comput. Sci. Inform. Technol. 6 (2) (2015) 1263–1266.
[26] D. Li, K. Kipper-Schuler, G. Savova, Conditional random fields and support vector machines for disorder named entity recognition in clinical texts, BioNLP 2008: Curr. Trends Biomedical Nat. Lang. Process. (2008) 94–95, https://doi.org/10.3115/1572306.1572326.
[27] J. Zhang, D. Shen, G. Zhou, J. Su, C.L. Tan, Enhancing HMM-based biomedical named entity recognition by studying special phenomena, J. Biomed. Inform. 37 (6) (2004) 411–422, https://doi.org/10.1016/j.jbi.2004.08.005.
[28] S.K. Saha, S. Sarkar, P. Mitra, Feature selection techniques for maximum entropy based biomedical named entity recognition, J. Biomed. Inform. 42 (5) (2009) 905–911, https://doi.org/10.1016/j.jbi.2008.12.012.
[29] K. Lee, Y. Hwang, S. Kim, H. Rim, Biomedical named entity recognition using two-phase model based on SVMs, J. Biomed. Informat. 37 (6) (2004) 436–447, https://doi.org/10.1016/j.jbi.2004.08.012.
[30] B. Tang, Y. Feng, X. Wang, et al., A comparison of conditional random fields and structured support vector machines for chemical entity recognition in biomedical literature, J. Cheminf. 7 (1) (2015) 232–240, https://doi.org/10.1186/1758-2946-7-S1-S8.
[31] B. Settles, Biomedical named entity recognition using conditional random fields and rich feature sets, in: JNLPBA '04 Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications, 2004, pp. 104–107.
[32] C. Sun, Y. Guan, X. Wang, L. Lin, Rich features based conditional random fields for biological named entities recognition, Comput. Biol. Med. 37 (9) (2007) 1327–1333, https://doi.org/10.1016/j.compbiomed.2006.12.002.
[33] V. Suarez-Paniagua, R.M.R. Zavala, I. Segura-Bedmar, P. Martinez, A two-stage deep learning approach for extracting entities and relationships from medical texts, J. Biomed. Inform. 99 (2019), 103285, https://doi.org/10.1016/j.jbi.2019.103285.
[34] M. Cho, J. Ha, C. Park, S. Park, Combinatorial feature embedding based on CNN and LSTM or biomedical named entity recognition, J. Biomed. Inform. 103 (2020), 103381, https://doi.org/10.1016/j.jbi.2020.103381.
[35] X. Wang, Y. Zhang, Y. Zhang, et al., Cross-type biomedical named entity recognition with deep multi-task learning, Bioinformatics 35 (10) (2019) 1745–1752, https://doi.org/10.1093/bioinformatics/bty869.
[36] W. Yoon, C.H. So, J. Lee, J. Kang, CollaboNet: collaboration of deep neural networks for biomedical named entity recognition, BMC Bioinf. 20 (2019) 249, https://doi.org/10.1186/s12859-019-2813-6.
[37] J. Lee, W. Yoon, S. Kim, et al., BioBERT: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (4) (2020) 1234–1240, https://doi.org/10.1093/bioinformatics/btz682.
[38] K. Xu, Z. Yang, P. Kang, Q. Wang, W. Liu, Document-level attention-based BiLSTM-CRF incorporating disease dictionary for disease named entity recognition, Comput. Biol. Med. 108 (2019) 122–132, https://doi.org/10.1016/j.compbiomed.2019.04.002.
[39] D.S. Sachan, P. Xie, M. Sachan, E.P. Xing, Effective use of bidirectional language modeling for transfer learning in biomedical named entity recognition, in: D.-V. Finale (Ed.), Proceedings of Machine Learning Research, Palo Alto, CA, vol. 85, 2018, pp. 383–402.
[40] K. Xu, Z. Zhou, T. Hao, W. Liu, A bidirectional LSTM and conditional random fields approach to medical named entity recognition, in: Proceedings of the International Conference on Advanced Intelligent Systems and Informatics; 2017 Sep 9-11; Cairo, Egypt, 2017, https://doi.org/10.1007/978-3-319-64861-3_33.
[41] L. Li, L. Jin, Z. Jiang, D. Song, D. Huang, Biomedical named entity recognition based on extended recurrent neural networks, in: IEEE International Conference on Bioinformatics and Biomedicine (BIBM); 2015 Nov 5-9; Washington, DC, USA, 2015, https://doi.org/10.1109/BIBM.2015.7359761.
[42] M. Habibi, L. Weber, M. Neves, D.L. Wiegandt, U. Leser, Deep learning with word embeddings improves biomedical named entity recognition, Bioinformatics 33 (14) (2017) 37–48, https://doi.org/10.1093/bioinformatics/btx228.
[43] Y. Wu, J. Xu, M. Jiang, Y. Zhang, H. Xu, A study of neural word embeddings for named entity recognition in clinical text. AMIA Annual Symposium Proceeding, 2015.
[44] L. Li, L. Jin, D. Huang, Recognizing biomedical named entities based on the sentence vector/twin word embeddings conditioned bidirectional LSTM, in: China National Conference on Chinese Computational Linguistics;2016 Oct 15-16, Springer International Publishing, Yantei, China 2016, 2016, https://doi.org/10.1007/978-3-319-47674-2_15.
[45] D. Huang, Z. Jiang, L. Zou, L. Li, Drug–drug interaction extraction from biomedical literature using support vector machine and long short term memory networks, Inf. Sci. 415–416 (2017) 100–109.
[46] S. Chowdhury, X. Dong, L. Qian, et al., A multitask bi-directional RNN model for named entity recognition on Chinese electronic medical records, BMC Bioinf. 19 (499) (2018), https://doi.org/10.1186/s12859-018-2467.
[47] J.M. Giorgi, G.D. Bader, Transfer learning for biomedical named entity recognition with neural networks, Bioinformatics 34 (23) (2018) 4087–4094.
[48] Z. Liu, M. Yang, X. Wang, et al., Entity recognition from clinical texts via recurrent neural network, BMC Med. Informat. Decision Mak. 17 (67) (2017), https://doi.org/10.1186/s12911-017-0468-7.
[49] C. Lyu, B. Chen, Y. Ren, D. Ji, Long short-term memory RNN for biomedical named entity recognition, BMC Bioinf. 18 (462) (2017), https://doi.org/10.1186/s12859-017-1868-5.
[50] M. Boden, A guide to recurrent neural networks and backpropagation, 2001.
[51] F.Z. El-Alami, S.O. El-Alaoui, N. En-Nahnahi, Deep neural models and retrofitting for Arabic text categorization, Int. J. Intell. Inf. Technol. 16 (2) (2020) 74–86.
[52] L. Bottou, Large-scale machine learning with stochastic gradient descent, Proc. COMPSTAT' (2010) 177–186, https://doi.org/10.1007/978-3-7908-2604-3_16.
[53] T.H. Dang, H.Q. Le, T.M. Nguyen, S.T. Vu, D3NER: biomedical named entity recognition using CRF-biLSTM improved with fine-tuned embeddings of various linguistic information, Bioinformatics 34 (20) (2018) 3539–3546, https://doi.org/10.1093//bioinformatics//bty356.
[54] B. James, B. Olivier, B. Frédéric, L. Pascal, P. Razvan, Theano: a CPU and GPU math expression compiler. Proceedings of the Python for Scientific Computing Conference (SciPy); 2010 June 28-30; Austin, Texas, 2010.
[55] P. Marimuthu, V. Perumal, V. Vijayakumar, Intelligent personalized abnormality detection for remote health monitoring, Int. J. Intell. Inf. Technol. 16 (2) (2020) 87–109.
[56] R.T.H. Tsai, S.H. Wu, W.C. Chou, et al., Various criteria in the evaluation of biomedical named entity recognition, BMC Bioinf. 7 (92) (2006), https://doi.org/10.1186/1471-2105-7-92.