A Hierarchical Model with Recurrent Convolutional Neural Networks for Sequential Sentence Classification

Xinyu Jiang[1], Bowen Zhang[2], Yunming Ye[1(✉)], and Zhenhua Liu[3[0000−0003−2760−3621]]

[1] School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China
[2] School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China
[3] NLP Group, Gridsum, Beijing, China
yym@hit.edu.cn

Abstract. Hierarchical neural networks approaches have achieved outstanding results in the latest sequential sentence classification research work. However, it is challenging for the model to consider both the local invariant features and word dependent information of the sentence. In this work, we concentrate on the sentence representation and context modeling components that influence the effects of the hierarchical architecture. We present a new approach called SR-RCNN to generate more precise sentence encoding which leverage complementary strength of bi-directional recurrent neural network and text convolutional neural network to capture contextual and literal relevance information. Afterwards, statement-level encoding vectors are modeled to capture the intrinsic relations within surrounding sentences. In addition, we explore the applicability of attention mechanisms and conditional random fields to the task. Our model advances sequential sentence classification in medical abstracts to new state-of-the-art performance.

Keywords: Sequential sentence classification · Hierarchical neural networks · Sentence representation.

1 Introduction

Text classification is an important task in many areas of natural language processing (NLP) which assigns pre-defined categories to free-text documents [26]. Most traditional high-performance text classification models are linear statistical models, including Naive Bayes[13], Support Vector Machine (SVM)[10, 33], Maximum Entropy Models[10], Hidden Markov Models (HMM)[23] and Conditional Random Fields (CRF)[11, 12, 16], which rely heavily on numerous carefully hand-engineered features. In recent years, non-linear deep neural networks (DNN) which do not require manual features are broadly applied to text classification with excellent result. Many approaches [6, 17, 21, 28, 35] are commonly based on convolutional neural network (CNN) or recurrent neural network

(RNN) or a combination of them. However, sentence-level texts are usually sequential (for example, sentences in a document or utterances in a dialog). The above-mentioned works do not take into account sequence order correlation characteristics of natural languages.

In order to distinguish from the general text or sentence classification, the categorization of sentences appearing in sequence is called the sequential sentence classification task [8].Specifically, the classification of each single sentence is related to the element categories of the surrounding sentences, which is different from the general sentence classification that does not involve context. This task is close to the sequence labeling task [14, 24, 30] which achieves assigning a categorical tag to each member of a series of observations and most approaches for implementing sequence labeling use bi-directional recurrent neural network (bi-RNN) and various extensions to the architecture.

There has been a considerable amount of work in text classification and sequence labeling, but much less work has been reported on the sequential sentence classification task, where categories of surrounding sentences have a great influence on the prediction of the central sentence. To the best of our knowledge, the first approach based on artificial neural network (ANN) to classify sequential short-text (dialog) was proposed by Lee and Dernoncourt [22], but only adding sequential information from preceding utterances. Subsequently, Dernoncourt et al. [9] presented a neural network structure based on both token and character embedding and used a CRF optimization layer to constrain the sequence results. Currently, the most influential approach obtained state-of-the-art results is due to Jin and Szolovits [15], in which the authors make use of the contextual information through adding a long short-term memory (LSTM) layer to processes encoded sentences. These works either introducing an RNN module or a CNN module to separately encode the sentence composition from the token embedding. However, on the one hand, the ability of CNN to extract local n-gram patterns depends on the fixed window size without considering the semantic relations and complicated syntactic of the sentence as a whole. On the other hand, RNN is able to capture tokens dependencies but ignore task-specific features on the feature vector dimension, which perhaps is essential for sentence representation.

In this paper, we present a novel hierarchical neural network architecture to tackle the sequential sentence classification task. Our model is mainly based on two critical components: sentence representation and context modeling. In the sentence representation component, we develop the SR-RCNN approach that is designed to capture both words hiding properties and sequential correlation features for producing the more precise sentences semantic. To benefit from the advantages of CNN and RNN, we first take the bi-directional recurrent structure that introduces appreciably less noise to keep a wider range of word sorting characters when learning the token embedding in a sentence. Second, we combine original word representation and transferred statement information as input to CNN for effectively model higher-level representation of sentences. In the context modeling component, we add the multilayer bi-RNN to enrich the semantic con-

textual information from preceding and succeeding sentences. In order to verify our ideas, we systematically analyze our model on two benchmarking datasets: NICTA-PIBOSO [16] and PubMed RCT [8]. Our main contributions can be summarized as follows:

1. We introduce a new neural network approach that relies on global and local grammatical patterns to model context relation between sentences for sequential sentence classification. Moreover, we consider two different alternative output strategies to predict the sequential labels.
2. Inspired by the previous best performing research work, we propose the SR-RCNN algorithm based on CNN and RNN which provide complementary linguistic information to optimize the sentence encoding vectors.
3. We report empirical results that the attention mechanism is more suitable for learning weights before the bi-RNN introduces text long-distance sequence features and whether adding a CRF layer to constraint the label sequence results depends on the specific dataset.
4. Our approach effectively improves the performance for sentence level classification in medical scientific abstracts when compared to the previous state-of-the-art methods.

2 Model

The major framework of our approach for sequential sentence classification is displayed in Figure 1. In this section, we will discuss each component (layer) of our neural network architecture in detail from bottom to top.

2.1 Word Representation

The bottom layer of the model structure is word representation, also known as word embedding [2], which maps tokens from discrete one-hot representations to dense real-valued vectors in a low-dimensional space. All the word vectors are stacked in the embedding matrix $W_{word} \in \mathbb{R}^{d \times |v|}$, where d is the dimension of the word vector and $|v|$ is the vocabulary of the dataset. Given a sentence comprising l words, we denote the sentence as $x = \{x_1, x_2, \ldots, x_l\}$, where x_i is the id of ith word in the vocabulary. Each sentence is converted into corresponding word vectors representation by embedding lookup. The embedding matrix W can be pre-trained from text corpus by embedding learning algorithms [25, 29, 3].

2.2 Sentence Encoding

CNN is good at extracting position-invariant features and RNN is able to flexibly model sequence dependencies for text classification[34]. We propose an algorithm that combines the capability of text-CNN [18] and bi-RNN [31] called SR-RCNN to enhance feature extraction and get the representation vector of the sentence. We fed the sentence embedding $w = \{w_1, w_2, \ldots, w_l\}$ into bi-RNN to automatically extracts the context-dependent features within a statement. Let $h =$

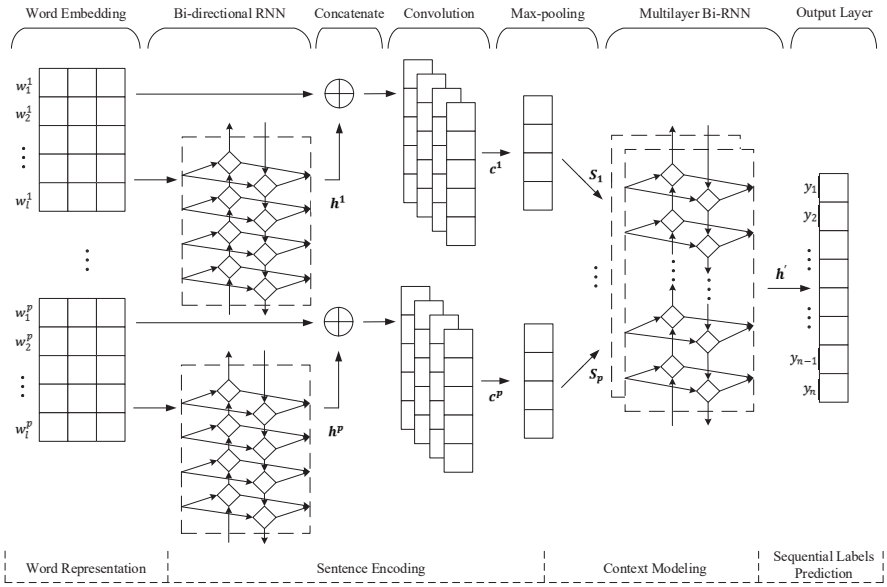

Fig. 1. The proposed neural network model architecture for sequential sentence classification.

$\{h_1, h_2, \ldots, h_l\}$ be the hidden representations output from the bi-RNN layer. Then, adding splicing $h_{1:l}$ and original word representation $w_{1:l}$ as the r input to the text-CNN to obtain more local features by one-dimensional convolution in the initial vocabulary information and the processed characteristics that introduces association dependencies . The feature map c_j is generated by:

$$c_j = \sigma(k_j \cdot r_{i:i+t-1} + b_j) \tag{1}$$

where $k_j \in \mathbb{R}^{t \times d}$ denotes a filter that convolutes a window of t words embedding $r_{i:i+t-1}$, and $b_j \in \mathbb{R}$ is a bias term. Here we use four filters with different window size and use ReLU [7] as the activation function σ to incorporate element-wise non-linearity. After that, we employ a max-overtime pooling operation [5] to capture the most important local features over the feature map.

2.3 Context Modeling

After the SR-RCNN component, we add multilayer bi-RNN to the model structure, using its powerful sequence modeling capabilities to capture long-term contextual information between sentences to enrich surrounding statements associated features. The sentence encodes $S = \{S_1, S_2, \ldots, S_p\}$ output by the SR-RCNN is convolved as an input to the multilayer bi-RNN. The bi-RNN takes into account both preceding histories (extracted by forward pass) and following evidence (extracted by backward pass). The output of multilayer bi-RNN

is hidden vectors, represented by $h' = \{h'_1, h'_2, \ldots, h'_l\}$, obtained by concatenating its forward and backward context representations. We convolve the output vectors, which can be regarded as the enriching sentence characteristics that take considerations of historical semantics with different granularities. Particularly, we utilize a convolutional layer to convert the context feature vector into a real-valued vector whose length is the number of categories, denoted as C.

2.4 Sequential Labels Prediction

Eventually, there are two output strategies for obtaining the label sequence. One is using the softmax layer to get the output directly, and the other is adding the CRF layer for label sequence optimization instead of modeling tagging decisions independently.

Softmax Adding a softmax layer for normalization to convert the true value to a relative probability between different categories, calculated as follows.

$$P_i = \frac{\exp(y_i)}{\sum_{i'=1}^{C} \exp(y_{i'})} \tag{2}$$

After the output layer of the neural network is subjected to the softmax function, the next step is to calculate the loss for model training. We use the cross-entropy as the loss function to calculate the error information between the predicted label sequence $P_i(a)$ and gold label sequence $P_i^g(a)$:

$$Loss = -\sum_{a \in D} \sum_{i=1}^{C} P_i^g(a) \cdot \log(P_i(a)) \tag{3}$$

where D is the training data, a represents an abstract, and P_i^g is the one-hot coding scheme of the tag list. When the classification is more correct, the dimension corresponding to the ground truth of P_i will be closer to 1 and the value of $Loss$ will be smaller. During the training phase, the objective is to minimize the cross-entropy loss which guides the network updating parameters through the back propagation.

CRF In the CRF layer[20], we introduce a labels transition matrix T, where $T_{i,j}$ denotes the transition probabilities of transition from label i to label j in successive sentences. This matrix is the parameter that needs to be trained in the CRF layer of model. It will learn the dependency constraints that may exist between successive tags. The output of the last multilayer bi-RNN is the probability sequence $P_{1:n}$ of n sentences, where $P_{i,j}$ indicates the probability that the jth label assigned to the ith sentence. Then the score of a prediction label sequence $y_{1:n}$ can be defined as:

$$S(y_{1:n}) = \sum_{i=2}^{n} T_{y_{i-1}, y_i} + \sum_{i=1}^{n} P_{i, y_i} \tag{4}$$

The conditional probability of a certain sequence is calculated using the softmax function by normalizing the above scores over all possible label sequences. During the training phase, the objective of the model is to maximize the log-probability of the gold label sequence. At inference time, the predicted labels sequence result is chosen as the one that obtains the maximum score. This can be calculated by the Viterbi algorithm[32].

3 Experiments

3.1 Datasets

We verify our proposed approach on two medical abstract datasets: PubMed RCT and NICTA-PIBOSO. Detailed statistics of two datasets are given in Table 1. $|C|$ represents the number of label categories and $|V|$ denotes the vocabulary size. For the train, validation and test datasets, the number of abstracts and the number of statements (in parentheses) are noted.

Table 1. Dataset statistics overview.

Dataset	$\|C\|$	$\|V\|$	Train	Validation	Test
PubMed 20k	5	68k	15k(180k)	2.5k(30k)	2.5k(30k)
PubMed 200k	5	331k	190k(2.2M)	2.5k(29k)	2.5k(29k)
NICTA	6	17k	800(8.6k)	-	200(2.2k)

PubMed RCT[4] is derived from PubMed database and provides two subsets: PubMed 20k and PubMed 200k [8]. It contains five classes: *objectives, background, methods, results* and *conclusions*.
NICTA-PIBOSO[5] released by Kim et al. [16] and was shared from the ALTA 2012 Shared Task [1]. The tag-set is defined as *background, population, intervention, outcome, study design* and *other*.

To offer a fair comparison with the best published results, all corpora have no other pre-processing operations except change to lower-cased. We did not remove any rare words and numbers from the corpora, resulting in a large number of tokens unrelated to the classification are delivered to the model. It is remarkable that our model still functioned well without any additional pre-processing.

3.2 Experimental Setting

During training, all word embeddings are initialized using the 'PubMed-and-PMC-w2v'[6] which were pre-trained on the corpus combining the publication

[4] The dataset is downloaded from: https://github.com/Franck-Dernoncourt/pubmed-rct
[5] https://www.kaggle.com/c/alta-nicta-challenge2
[6] The word vectors are downloaded from: http://evexdb.org/pmresources/vec-space-models/

abstracts from PubMed and the full-text articles from PubMed Central (PMC) open access subset [27] using the word2vec tool with 200 dimensions. The word vectors are fixed during the training phase. In order to get the best performance from the model, we have also tried other word embeddings, but there were no obvious benefits.

The hyperparameter settings of the model on both datasets are described below, all of which are selected by altering one each time while keeping other hyperparameters unchanged. For SR-RCNN module, the hidden layer of bi-RNN has size 128 and the filter windows of text CNN are designed to $c = (1, 2, 3, 4)$ with 256 filters each. For context modeling module, the hidden layer is set to 256 dimensions in multilayer bi-RNN. And the type of the recurrent unit defaults to gated recurrent unit (GRU) in the bi-RNN layer. For optimization, parameters are trained using Adam [19]. In order to accelerate the training process, the model uses batch-wise training of 40 abstracts per batch (for PubMed dataset, 16 for NICTA dataset) and we truncate or zero-pad sentences to ensure that each sentence is 60 tokens in length. Besides, we apply dropout on both the input and output vectors of bi-RNN to mitigate over-fitting. The dropout rate is fixed at 0.8 for all dropout layers through all the experiments.

Previous works relied mainly on F1-score to evaluate system performance, so we also provide F-score as the evaluation indicator for better comparison with existing literature.

4 Results and Discussion

4.1 Comparison with Other Works

We compare our results with several baselines as well as recent state-of-the-art works results on the three datasets. As shown in Table 2, our model performs best on all datasets, promoting previous best published results by 0.5%, 0.5% and 2.5% on the PubMed 20k, PubMed 200k and NICTA-PIBOSO dataset, respectively. It proves that the hierarchical model framework based on SR-RCNN and multilayer bi-RNN is effective in solving the sequential sentence classification task.

Compared to other systems, our approach that automatically learns numerous features from the context does not require careful hand-engineered features. Analyzing the promotion scores for three datasets, NICTA-PIBOSO is the smallest dataset but has the highest improvement. Our approach is applicable to small data without over-fitting. Besides, for the results of PubMed RCT datasets, our model offers better performance with sufficient data to adjust parameters. We also compare two different types of gated units in bi-RNN. Because more sophisticated recurrent units are indeed better than more traditional recurrent units [4], we focus on LSTM and GRU to implement gating mechanisms. These results indicate that applying GRU in the bi-rnn layers brings better performance on these three abstracts datasets for sequential sentence classification task.

Furthermore, our model is not only effective but also efficient. Experiments are performed on a GeForce GTX 1080 Ti GPU and the entire training procedure

Table 2. Experimental comparison results with other seven models.

Model	PubMed 20k	PubMed 200k	NICTA
Logistic Regression	83.1	85.9	71.6
Forward ANN	86.1	88.4	75.1
CRF	89.5	91.5	81.2
Best published[1]	-	-	82.0
bi-ANN [9]	90.0	91.6	82.7
HSLN-CNN [15]	92.2	92.8	84.7
HSLN-RNN [15]	92.6	93.9	84.3
Our Model(LSTM)	92.9	94.1	86.8
Our Model(GRU)	**93.1**	**94.4**	**87.2**

(on the PubMed 20k dataset) takes about half an hour with 7 epochs to achieve the best result.

4.2 Model Analysis

We conduct refined experiments to analyze and separate out the contribution of each component of our model. We compared the effectiveness of models variants combination of different sentence representation and context modeling components in the medical scientific abstracts. The overview of the comparison results of all the models is shown in Table 3. '+Att.' indicates that we directly apply the attention mechanism (AM) on the sentence representations. The sentences encoding vectors output from the attention are the weighted sum of all the input. 'n-l' means n layers.

Table 3. Model performance comparison results with different components.

Sentence Representation	Context Modeling	PubMed 20k	PubMed 200k	NICTA
Text CNN	1-l bi-RNN	92.2	94.1	85.2
Text CNN	1-l bi-RNN + CRF	92.3	94.0	85.6
Text CNN + Att.	1-l bi-RNN	92.3	94.1	85.4
Text CNN + Att.	1-l bi-RNN + CRF	92.4	94.1	85.7
SR-RCNN	1-l bi-RNN	93.0	94.2	**87.2**
SR-RCNN	1-l bi-RNN + CRF	92.8	94.2	86.8
SR-RCNN + Att.	1-l bi-RNN	92.9	94.1	86.1
SR-RCNN + Att.	1-l bi-RNN + CRF	92.7	94.0	86.1
SR-RCNN	2-l bi-RNN	**93.1**	**94.4**	86.7
SR-RCNN	2-l bi-RNN + CRF	92.9	**94.4**	86.6
SR-RCNN + Att.	2-l bi-RNN	93.0	94.2	87.0
SR-RCNN + Att.	2-l bi-RNN + CRF	92.8	94.2	86.7

According to the results in Table 3, the performance of the model is consistently improved by using our SR-RCNN method instead of text CNN as sentence representation on all the datasets. It proves that the collaborative component is a good blend of the advantages of the bi-RNN and text CNN for sentence encoding. Moreover, the scores of models with one layer bi-RNN or two layers bi-RNN in PubMed datasets indicate that increasing the number of bi-RNN layers can enhance the capability of the model to enrich contextual features, thereby further improving the classification quality of the model. However, the models with two layers bi-RNN based context modeling are not as good as the models with one layer bi-RNN on the NICTA dataset. This is due to the dataset smallest scale, so it is easier to over-fitting in more complex neural network architecture. In addition, we have tried to apply the AM to the pooling layer of text CNN, which contributes to the overall improvements before introducing SR-RCNN architecture. Our intuition is that in abstracts the dependency between the sequential sentences should be weighted before being processed by the bi-RNN.

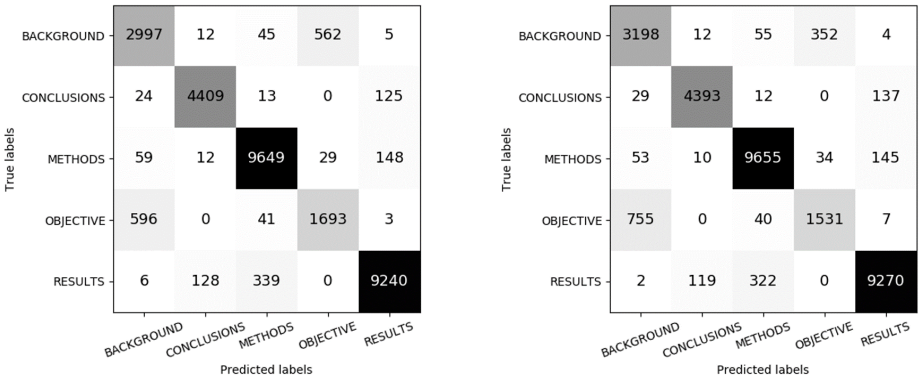

Fig. 2. Confusion matrices on the PubMed 20k test dataset achieved by our model with CRF (matrix on the left) and without CRF (matrix on the right).

Comparing the two structures of the label results directly predicted by the softmax layer and the results of further optimization with the CRF layer, the impact of CRF on the accuracy of the model results is uncertain. For further detailed inspecting of the reasons for the deterioration of the experimental results, taking the model with SR-RCNN and two layers bi-RNN as an example, we list the Figure 2 which is the confusion matrices of the test results achieved by our model with CRF layer and without CRF layer respectively. Checking every specific value of each matrix, we detect that the biggest gap between two matrices is on the *background* and *objective* labels. For finding the cause of the difference, we examined the transition matrix obtained by the CRF layer which encodes the transition probability between two subsequent labels, as shown in

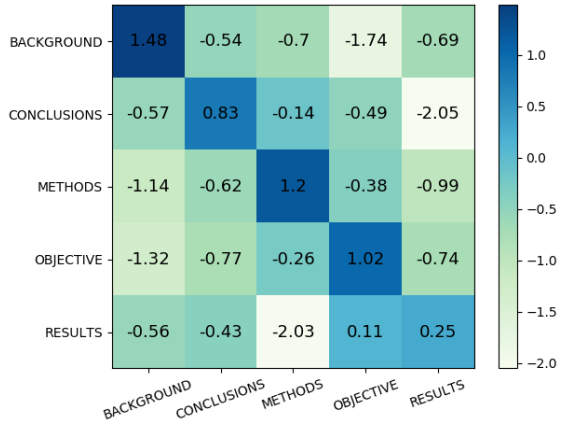

Fig. 3. Transition matrix of CRF layer learned on PubMed 20k dataset.

Figure 3. In the transition matrix, the columns display the current sentence tag and the rows display the previous sentence tag. The *methods* class has the largest percentage of all the sentence classes (9897 of 30135 32.84%). We can conclude that a sentence pertaining to *objective* is more likely to be followed by a sentence pertaining to *method* (with a smaller penalty, -0.7<-0.26) than a sentence pertaining to *background* through the transfer matrix. The model with CRF layer is more inclined to predict sentences as *objective* than *background* accordingly. Due to the imbalance of samples (the number of *background* sentences is more than the *objective* sentences), the model without CRF is more inclined to predict the sentence as the *background* (labels with more samples). Not using the CRF layer to add constraints to the final predicted labels on these medical scientific abstracts sets has a better effect, but more experiments are still needed for other specific datasets and related tasks.

5 Conclusion

In this paper, we explore the factors that affect the performance of hierarchical neural networks for sequential sentences classification. Our experiment shows that a stronger sentence level feature extractor (SR-RCNN) and a well sequential feature fusion operator (multilayer bi-RNN) are the key factors that improve the model performance. While the attention mechanism and the CRF optimization layer may shift the results, the overall effect is uncertain. Our results were confirmed on two benchmarking datasets of medical scientific abstracts with state-of-the-art results, which provide a basic guidance for further research and practical application.

Acknowledgment

This research was supported in part by NSFC under Grant No.U1836107 and No.61572158.

References

1. Amini, I., Martinez, D., Molla, D., et al.: Overview of the alta 2012 shared task (2012)
2. Bengio, Y., Ducharme, R., Vincent, P., Jauvin, C.: A neural probabilistic language model. Journal of machine learning research **3**(Feb), 1137–1155 (2003)
3. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics **5**, 135–146 (2017)
4. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014)
5. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. Journal of machine learning research **12**(Aug), 2493–2537 (2011)
6. Conneau, A., Schwenk, H., Barrault, L., Lecun, Y.: Very deep convolutional networks for text classification. arXiv preprint arXiv:1606.01781 (2016)
7. Dahl, G.E., Sainath, T.N., Hinton, G.E.: Improving deep neural networks for lvcsr using rectified linear units and dropout. In: 2013 IEEE international conference on acoustics, speech and signal processing. pp. 8609–8613. IEEE (2013)
8. Dernoncourt, F., Lee, J.Y.: Pubmed 200k rct: a dataset for sequential sentence classification in medical abstracts. arXiv preprint arXiv:1710.06071 (2017)
9. Dernoncourt, F., Lee, J.Y., Szolovits, P.: Neural networks for joint sentence classification in medical paper abstracts. arXiv preprint arXiv:1612.05251 (2016)
10. Hachey, B., Grover, C.: Sequence modelling for sentence classification in a legal summarisation system. In: Proceedings of the 2005 ACM symposium on Applied computing. pp. 292–296. ACM (2005)
11. Hassanzadeh, H., Groza, T., Hunter, J.: Identifying scientific artefacts in biomedical literature: The evidence based medicine use case. Journal of Biomedical Informatics **49**, 159 – 170 (2014)
12. Hirohata, K., Okazaki, N., Ananiadou, S., Ishizuka, M.: Identifying sections in scientific abstracts using conditional random fields. In: Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I (2008)
13. Huang, K.C., Chiang, I.J., Xiao, F., Liao, C.C., Liu, C.C.H., Wong, J.M.: Pico element detection in medical text without metadata: Are first sentences enough? Journal of biomedical informatics **46**(5), 940–946 (2013)
14. Jagannatha, A.N., Yu, H.: Structured prediction models for rnn based sequence labeling in clinical text. In: Proceedings of the conference on empirical methods in natural language processing. conference on empirical methods in natural language processing. vol. 2016, p. 856. NIH Public Access (2016)
15. Jin, D., Szolovits, P.: Hierarchical neural networks for sequential sentence classification in medical scientific abstracts. arXiv preprint arXiv:1808.06161 (2018)
16. Kim, S.N., Martinez, D., Cavedon, L., Yencken, L.: Automatic classification of sentences to support evidence based medicine. In: BMC bioinformatics. vol. 12, p. S5. BioMed Central (2011)

17. Kim, T., Yang, J.: Abstractive text classification using sequence-to-convolution neural networks. arXiv preprint arXiv:1805.07745 (2018)
18. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
19. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
20. Lafferty, J., McCallum, A., Pereira, F.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: Proc. 18th International Conf. on Machine Learning. pp. 282–289 (2001)
21. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: Twenty-ninth AAAI conference on artificial intelligence (2015)
22. Lee, J.Y., Dernoncourt, F.: Sequential short-text classification with recurrent and convolutional neural networks. arXiv preprint arXiv:1603.03827 (2016)
23. Lin, J., Karakos, D., Demner-Fushman, D., Khudanpur, S.: Generative content models for structural analysis of medical abstracts. In: Proceedings of the HLT-NAACL BioNLP Workshop on Linking Natural Language and Biology. pp. 65–72. LNLBioNLP '06, Association for Computational Linguistics, Stroudsburg, PA, USA (2006)
24. Liu, L., Shang, J., Ren, X., Xu, F.F., Gui, H., Peng, J., Han, J.: Empower sequence labeling with task-aware neural language model. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
25. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013)
26. Mirończuk, M.M., Protasiewicz, J.: A recent overview of the state-of-the-art elements of text classification. Expert Systems with Applications **106**, 36–54 (2018)
27. Moen, S., Ananiadou, T.S.S.: Distributional semantics resources for biomedical text processing. Proceedings of LBM pp. 39–44 (2013)
28. Moriya, S., Shibata, C.: Transfer learning method for very deep cnn for text classification and methods for its evaluation. In: 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC). vol. 2, pp. 153–158. IEEE (2018)
29. Pennington, J., Socher, R., Manning, C.: Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 1532–1543 (2014)
30. Reimers, N., Gurevych, I.: Optimal hyperparameters for deep lstm-networks for sequence labeling tasks. arXiv preprint arXiv:1707.06799 (2017)
31. Schuster, M., Paliwal, K.K.: Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing **45**(11), 2673–2681 (1997)
32. Viterbi, A.: Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE transactions on Information Theory **13**(2), 260–269 (1967)
33. Yamamoto, Y., Takagi, T.: A sentence classification system for multi biomedical literature summarization. In: 21st International Conference on Data Engineering Workshops (ICDEW'05). pp. 1163–1163 (April 2005)
34. Yin, W., Kann, K., Yu, M., Schuetze, H.: Comparative study of cnn and rnn for natural language processing (2017). arXiv preprint arXiv:1702.01923 (2017)
35. Zhou, Y., Xu, B., Xu, J., Yang, L., Li, C.: Compositional recurrent neural networks for chinese short text classification. In: 2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI). pp. 137–144. IEEE (2016)