Fully Contextualized Biomedical NER

Abstract. Recently, neural network architectures have outperformed traditional methods in biomedical named entity recognition. Borrowed from innovations in general text NER, these models fail to address two important problems of polysemy and usage of acronyms across biomedical text. We hypothesize that using a fully-contextualized model that uses contextualized representations along with context dependent transition scores in CRF can alleviate this issue and help further boost the tagger's performance. Our experiments with this architecture have shown to improve state-of-the-art F1 score on 3 widely used biomedical corpora for NER. We also perform analysis to understand the specific cases where a contextualized model is superior to a strong baseline. We will make implementation of our paper available upon acceptance of the paper.

1 Introduction

Biomedical Named Entity Recognition (NER) is a fundamental step in several downstream biomedical text mining and information extraction tasks like relation classification, co-reference resolution etc. Traditional Biomedical NER systems [7, 8] have often relied on task specific hand crafted features. Recent neural network based architectures in biomedical domain [15] have shown that comparable results can be achieved without making use of these hand engineered features although the performance is still significantly dependent on the quality of learned word representations [16]. Character embeddings and pre-trained distributed embeddings have been used to model complex syntactic and semantic characteristics of words. But these complementary embedding models fail to capture different word uses across different linguistic contexts (i.e. *polysemy*). This problem is compounded in biomedical text due to ambiguous usage of words from general text [14] (ex: *column* in general english means *an upright pillar* while in medical context can be taken to mean *the spine*). Word representations obtained from training on biomedical corpora do not solve this problem because both forms, general english and biomedical, are generally present in the training text.

Another issue specific to biomedical domain is the generous usage of abbreviations (ex: gene/protein names like *ALA, MEN 1*) without explicit mention of their full forms. Neither character embeddings nor distributed word embeddings are effective in solving this issue. Character embeddings do not help as these abbreviations are mostly acronyms, where all characters are capitalized irrespective of the entity type. Word embeddings generally fail as most of these acronyms fall outside their vocabulary.

In order to ameliorate these two issues, we look at contextualization as an alternative. Current biomedical NER systems make use of context with the help of a Bi-directional LSTM that sequentially processes a sentence [16, 18, 15]. Our

model captures context more effectively in two additional ways: First, we make use of contextualized word representations based on [13], which have shown to improve sequence tagging with general english text [12, 13]. Following the earlier discussion on issues with current biomedical NER systems, we find that these contextualized word representations are especially helpful in biomedical domain. Second, for the CRF layer we use a context dependent transition matrix [3] which is conditioned on token as well as its immediate context. We model this transition matrix non-linearly with the help of different neural networks. Experiments on four widely used biomedical datasets show that we are able to obtain state-of-the-art performance using this fully contextualized NER tagger.

Our main contributions are summarized as : (1) We show that using contextualized word embeddings for Biomedical NER leads to better performance in comparison to the baseline system. (2) We explore the use of different types of neural networks to model pairwise transition scores for CRF which further improves the tagging performance. (3) Our proposed model provides an improvement over current state-of-the-art performance in 3 out of 4 standard biomedical NER datasets.

2 Proposed Method

Our overall neural network architecture is shown in Figure 1 which uses a Bi-directional LSTM with a CRF sequence layer stacked on top of it. The input to this model is obtained by concatenating pre-trained distributed word representations with contextualized embeddings from a bi-directional LM. In sequence labeling tasks like NER, a standard conditional random field (CRF) [5] layer on top of a Bi-LSTM network is used to jointly model the dependency across final output labels [2, 6, 15]. Following [2, 6], a score over the output sequence $\mathbf{y} = (\mathrm{y_1}, \mathrm{y_2}, \ldots, \mathrm{y_n})$ is computed by summation of unary scores and pairwise transition scores as :

$$s(\mathbf{X}, \mathbf{y}) = \sum_{i=0}^{n} A_{\mathrm{y_i}, \mathrm{y_{i+1}}} + \sum_{i=1}^{n} P_{\mathrm{i}, \mathrm{y_i}} \tag{1}$$

where $\mathbf{P}$, a $n \times k$ matrix is used to model the unary scores and $\mathbf{A}$, a $k \times k$ matrix is used for transition scores, with n being the sequence length and k being the size of the tag set. The training is done with backpropagation by minimizing the negative log-likelihood of the correct label sequence $\hat{\mathbf{y}}$ for input $\mathbf{X}$ as follows:

$$-\log(p(\hat{\mathbf{y}}|\mathbf{X})) = -\log\left(\frac{e^{s(\mathbf{X}, \hat{\mathbf{y}})}}{\sum_{\mathbf{y} \in \boldsymbol{Y}} e^{s(\mathbf{X}, \mathbf{y})}}\right) \tag{2}$$

During inference, Viterbi algorithm is used for determining the final label sequence.

Contextualized Word Representations: Recently, contextualized word vectors have shown to improve performance in many downstream tasks [11–13]. [13] show that contextualized word vectors obtained from a bi-directional language model achieve state of the art results on NER in english domain. They use a CNN with varying filter sizes over characters and use a 2 layer Bi-directional model. Finally, they compute a linear combination over hidden states stacked

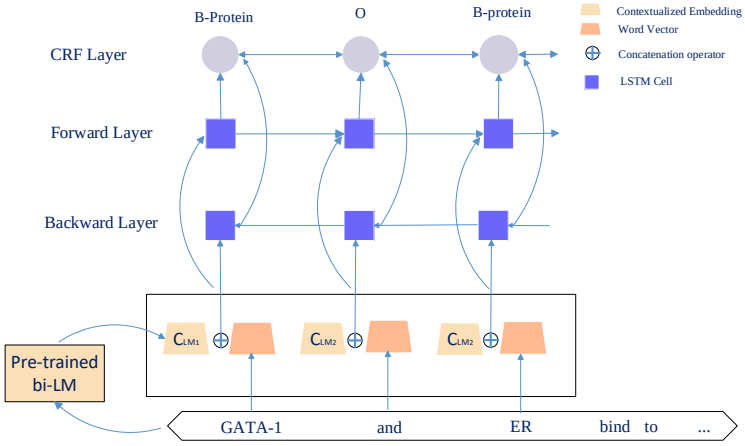

Fig. 1: Overview of the sequence tagging architecture. The word vector from pre-trained skip-gram model is concatenated with contextualized embeddings from a 2 layer bi-directional language model. See text for details.

on each token to get a final word representation, which they call as ELMo.

In order to deal with issue of polysemy and acronyms in biomedical text, we use contextualized embeddings from an architecture similar to ELMo by training it on biomedical text with 1 Billion tokens obtained from PubMed and PMC. A key difference is that we do not take linear combination of the hidden state vectors from each layer (as is done with ELMo) and rather simply concatenate them as we observe that doing this results in a slight decrease in performance on the development set. We will make this trained biomedical language model publicly available for future research after acceptance of paper.

Pairwise modeling with Neural Networks: Most neural linear-chain CRF models for sequence tagging fix the transition matrix, $\mathbf{A}$, after the training process has concluded. In case of rare medical entities, this parameter matrix might not effectively model the transitions between the labels [3]. We, therefore, study two different neural-network based methods to non-linearly model these transition parameters conditioned on the current token and its context.

We keep unary scores for the Bi-LSTM-CRF unchanged and denote the pairwise scores modeled by a neural network by $\phi_{nn}(\mathbf{X})_{i,i+1}$, where i denotes the label position in the sequence. For modeling these scores, we consider two different types of neural networks, namely, a Fully Connected (FC) Feed Forward Neural Network and a Convolutional Neural Network (CNN). For each transition, the neural network takes as input the feature representation from Bi-LSTM of the two neighboring tokens involved in transition and outputs a $k \times k$ matrix of transition scores. So for a sentence with length n, the neural network outputs a tensor of shape $n \times k \times k$. The training objective remains the same and is trained end-to-end corresponding to (2).

Table 1: Dataset Statistics

Dataset	Entity Types	Sentences (Train/Dev/Test)	# Mentions
JNLPBA[4]	Protein, DNA, RNA Cell Line, Cell Type	24,806 (18,607/1,939/4,260)	59,973
BC2GM[17]	Protein/Gene	20,000 (10,000/5,000/5,000)	24,550
BC5CDR[9]	Disease, Chemical	13,938 (4,560/4,581/4,797)	28,785
NCBI-Disease[1]	Disease	7,295 (5,432/923/940)	6,892

3 Datasets and Experimental Setup

Datasets: We use 4 widely used biomedical NER datasets to validate our method. Across these 4 datasets, all important biomedical entity types are covered. Statistics regarding mentions in corpora are mentioned in Table 1. We perform *exact match* (both entity type and entity boundary should be correctly predicted) evaluation based on macro-averaged F1 scores on all of these datasets.
Training: As input to our system, we concatenate distributed word vectors with contextualized word representations. Instead of providing the contextualized embeddings at the input, we also try concatenating them before the CRF layer but find that it performs marginally worse. We provide the necessary training and implementation details in the supplementary material [1].

4 Results and Discussions

4.1 Evaluation of architectures for pairwise modeling

We first evaluate different neural network architectures for modeling pairwise scores. We explore two prominent neural network architectures in CNN and Fully Connected NN (FCNN). In case of CNN, we perform one dimensional convolution with filter width 2 along with tanh non-linearity. For every sequence with length n, the CNN performs a 1-D convolution for each of $n-1$ transitions possible so that no padding is required. We also try a multi-layer CNN where we perform padding for second layer convolutions to obtain scores for $n-1$ transitions involved. In case of FCNN, we element wise multiply the feature representation for the two tokens involved in the transition. We experiment with a single layer FCNN and a multi layer FCNN with depth 2. We observe that using a CNN for modeling transitions generally performs slightly better than FCNN. We find that FCNN with 2 layers can provide slightly better precision. For each of the datasets, we choose the pairwise modeling technique that performs best on development set. We find that a CNN with depth 1 works best for all datasets except for BC2GM, where CNN with depth 2 performs better. The detailed results are provided in supplementary material.

4.2 Comparison with state of the art methods

We compare our results with other neural network methods known to perform well on these datasets. As a baseline we implement the Bi-LSTM-CRF tagger described in [6] which makes use of pre-trained embeddings and incorporates

[1] https://www.dropbox.com/s/zc53mw8n77aop27/SupplementaryMaterial.pdf?dl=0

Table 2: Comparison with baseline and state of the art method, based on F1-score. *: Our implementation. **: From paper

Model	NCBI-Dis	BC2GM	BC5CDR	JNLPBA
Lample et .al (2016)[6]*	85.59	79.35	86.12	73.75
Ma et al. (2016)[10]*	83.70	77.92	85.95	72.72
Deep Multi-Task (2018)[18]**	86.14	80.74	**88.78**	73.52
Proposed Model	**88.31**	**82.06**	**88.64**	**76.20**

subword features using another Bi-LSTM. We implement another baseline that instead uses CNN for calculating character embeddings[10]. Finally, we compare our method with a deep multi-task model, which provides state of the art performance on these datasets[18]. For the baseline, we performed hyper-parameter tuning with LSTM cell size and dimension of pre-trained word vectors. We did not perform any hyper-parameter tuning for our model. (Refer Suppl. material)

We observe that our proposed model outperforms the two baselines significantly on all 4 datasets (Table 2). In comparison to the Multi-Task model(MTM), our model is superior in performance by more than 1 percent on 3 out of 4 datasets. Likely reason the MTM slightly outperforms our model on BC5CDR dataset is that MTM uses 3 different datasets that have either one of the Chemical and Disease entities, in which case using a MTM seems to have helped.

Table 3: Ablations with F1-score on test set. *: Instead use CharLSTM (See Text) to make a fair comparison

Model	NCBI-Dis	BC2GM	BC5CDR	JNLPBA
Full Model	88.31	82.06	88.64	76.20
- NN Transition Scores	87.05	80.96	88.21	75.53
- Contextualized*	86.12	79.92	86.67	74.23

4.3 Ablations

To highlight the importance of the two important components in our model, we perform ablation analysis on the final test set without changing any hyper-parameters (see Table 3). First, we remove the contextualized embeddings from our model but incorporate LSTM based sub-word features (like the baseline) along with context-dependent NN based transition matrix. In the second case, we use contextualized embeddings with pre-trained word vectors and replace the context dependent transition matrix with a fixed (after training) transition matrix. We mainly observe that : (1) The contribution of contextualized word embeddings is more prominent and leads to an increase of almost 2 % in all cases. (2) When we remove the context dependent NN based transition score, in 3 out of 4 cases, F1 score drops by more than 1 %.

4.4 Understanding the effect of contextualized representations

To gain more insights into our proposed model, in particular the importance of contextualized representations in biomedical text, we select some examples from

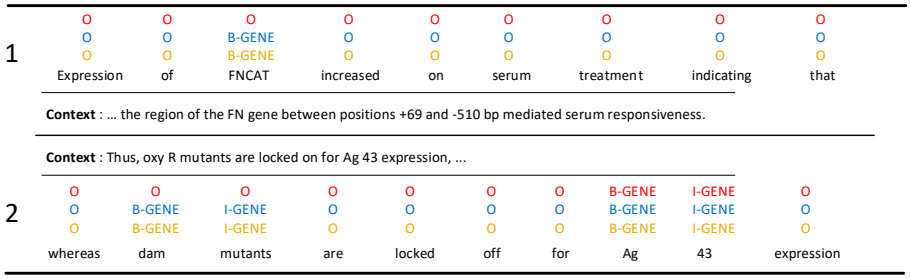

Fig. 2: Example outputs of our proposed model in cases where context helps determine the tags. Example 1 shows a case where the token is an acronym and Example 2 is a case of *polysemy* (for token *dam*). Entity tags highlighted with gold are the gold-standard tags, with red are the ones from baseline, and with blue, are from our proposed model. Related context is also shown.

the labeled test set. In figure 2, two recurring cases where using contextualized word embeddings have helped are shown. Example 1 exhibits the behavior of the baseline [6] and our proposed model in case of an acronym, the usage of which is very common in scientific texts. Using character representations, which might capture the information that the token is capitalized, alone does not help as the token *FNCAT* is out of vocabulary for pre-trained embeddings. It is interesting to note that a bi-directional LSTM, which is supposed to make use of the context, has not helped here either. This mistake is rectified by using a fully contextualized model, which looks at the neighboring token *Expression*, and infers that the entity involved (*FNCAT*) is a gene/protein.

Looking at example 2, we understand how using contextualized embeddings help to deal with polysemy. The token *dam*, which is more commonly associated with a reservoir structure, is incorrectly labeled by the baseline model. The baseline system correctly labeled *oxy R mutants* as a gene/protein entity but did not recognize *dam mutants*. Looking at the context suggests that like *oxy R mutants*, *dam mutants* should also be a protein. Again in such cases, looking at the larger context might have helped our model.

Finally, on analysis, we find that a fully contextualized model like ours also does much better on longer entities. We find that for entities with size greater than 5, our proposed model outperforms the baseline by a bigger margin. Detailed plots are available in supplementary material.

5 Conclusions

In this paper, we proposed a fully contextualized NER architecture that makes use of context more effectively by using contextualized representations along with context conditioned transition scores. Our proposed model significantly outperformed the baseline on all the datasets. In addition, our experiments have shown to beat current state-of-the-art results on 3 out of 4 datasets. All of our results were achieved without tuning hyper-parameters to specific datasets. Our detailed analysis in section 4.4 indicates why in the case of acronyms and polysemy, using a fully contextualized model might have helped.

References

1. Doğan, R.I., Leaman, R., Lu, Z.: Ncbi disease corpus: a resource for disease name recognition and concept normalization. Journal of biomedical informatics **47**, 1–10 (2014)
2. Huang, Z., Xu, W., Yu, K.: Bidirectional lstm-crf models for sequence tagging. arXiv preprint arXiv:1508.01991 (2015)
3. 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)
4. Kim, J.D., Ohta, T., Tsuruoka, Y., Tateisi, Y., Collier, N.: Introduction to the bio-entity recognition task at jnlpba. In: Proceedings of the international joint workshop on natural language processing in biomedicine and its applications. pp. 70–75. Association for Computational Linguistics (2004)
5. Lafferty, J., McCallum, A., Pereira, F.C.: Conditional random fields: Probabilistic models for segmenting and labeling sequence data (2001)
6. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. arXiv preprint arXiv:1603.01360 (2016)
7. Leaman, R., Gonzalez, G.: Banner: an executable survey of advances in biomedical named entity recognition. In: Biocomputing 2008, pp. 652–663. World Scientific (2008)
8. Leaman, R., Islamaj Doğan, R., Lu, Z.: Dnorm: disease name normalization with pairwise learning to rank. Bioinformatics **29**(22), 2909–2917 (2013)
9. Li, J., Sun, Y., Johnson, R.J., Sciaky, D., Wei, C.H., Leaman, R., Davis, A.P., Mattingly, C.J., Wiegers, T.C., Lu, Z.: Biocreative v cdr task corpus: a resource for chemical disease relation extraction. Database **2016** (2016)
10. Ma, X., Hovy, E.: End-to-end sequence labeling via bi-directional lstm-cnns-crf. arXiv preprint arXiv:1603.01354 (2016)
11. McCann, B., Bradbury, J., Xiong, C., Socher, R.: Learned in translation: Contextualized word vectors. In: Advances in Neural Information Processing Systems. pp. 6294–6305 (2017)
12. Peters, M.E., Ammar, W., Bhagavatula, C., Power, R.: Semi-supervised sequence tagging with bidirectional language models. arXiv preprint arXiv:1705.00108 (2017)
13. Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. arXiv preprint arXiv:1802.05365 (2018)
14. Pisanelli, D.M., Gangemi, A., Battaglia, M., Catenacci, C.: Coping with medical polysemy in the semantic web: The role of ontologies. In: Medinfo. pp. 416–419 (2004)
15. Sahu, S.K., Anand, A.: Recurrent neural network models for disease name recognition using domain invariant features. arXiv preprint arXiv:1606.09371 (2016)
16. Sahu, S.K., Anand, A.: Unified neural architecture for drug, disease and clinical entity recognition. arXiv preprint arXiv:1708.03447 (2017)
17. 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.: Overview of biocreative ii gene mention recognition. Genome biology **9**(2), S2 (2008)
18. Wang, X., Zhang, Y., Ren, X., Zhang, Y., Zitnik, M., Shang, J., Langlotz, C., Han, J.: Cross-type biomedical named entity recognition with deep multi-task learning. arXiv preprint arXiv:1801.09851 (2018)