CME² Net: Contextual Medical Event Extraction Network for clinical notes

Aman Sinha[1,2,†], Ayan Vishwakarma[3,†], Marianne Clausel[1] and Mathieu Constant[2]

[1]IECL, Université de Lorraine, Nancy, France
[2]ATILF, Université de Lorraine, Nancy, France
[3]Indian Institute of Technology Dhanbad, Jharkhand, India

Abstract
Medication change is very important to know the medical history of a patient. Most of the clinical notes are in unstructured format and in addition to that due to its narrative nature expert human annotators are needed to interpret the events, which is quite expensive. In this work, we present an end-to-end model for the task of automatic extracting and classifying the medication change events from a clinical note. We propose a joint learning model trained with adaptive sample weighting loss which incorporates the use of clinical contextual embedding and static embeddings. Our proposed system obtained competitive performance on CMED dataset (n2c2 challenge 2022) for contextual medical event detection and classification.

Keywords
Electronic Health Record, Medical Event Extraction, Event Classification

1. Introduction

Clinical NLP has benefited by the advancement in the field of Natural Language Processing and Information Retrieval. Several shared tasks involving automatic extraction and annotations of medical concepts and events have been organized by BioCreative [1], and NLP clinical Challenges (n2c2) in the past years to encourage the research on clinical textual data such as Electronic health records (EHRs). EHRs contains unstructured data which are a rich source of information and are essential to design custom healthcare pathways for precision medicine. The narrative nature of EHRs often make complicates the extraction and classification of clinical events. The arrival of transformer [1] architectures has provided new directions on clinical NLP applications allowing to extract finer contextual aspects of events in these notes.

The n2c2 2022 challenge[2] track-1 problem statement consisted of 3 tasks, namely (a) medication extraction, (b) event classification and (c) context classification. For a given clinical note, the system is required to extract any mentioned medicine name, then classify whether it is associated to a `medication_change` referred to as *Disposition* (otherwise *NoDisposition* or

[†]These authors contributed equally.
aman.sinha@univ-lorraine.fr (A. Sinha); ayanvishwakarma1248.19je0209@mc.iitism.ac.in (A. Vishwakarma); marianne.clausel@univ-lorraine.fr (M. Clausel); mathieu.constant@univ-lorraine.fr (M. Constant)

[1]https://biocreative.bioinformatics.udel.edu/
[2]https://n2c2.dbmi.hms.harvard.edu/

Undetermined), in which case it has to further classify the context of the medication change event based on the following longitudinal dimensions: `Action`, `Negation`, `Actor`, `Temporality` and `Certainty`.

2. Related Works

Several approaches including machine learning [2, 3] and deep learning [4, 5] have been explored to medical entities extraction and context classification.

Rule-based [6] and machine learning based models [2, 3] such as Decision Tree, Naïve Bayes, and SVM encounter difficulties because of out-of-vocabulary entities, unbalanced datasets, indirect state changes and subtle difference between different class definitions. While, deep-learning based methods such as hybrid model using RNNs and residual network [4]; multitask learning [5] suffer mostly ambiguity caused by writing styles (such as misspellings, abbreviations, inconsistent tense usage) in EHRs.

We, therefore, propose CME^2 network for extracting and identifying medication change event by using static and contextual embeddings to incorporate domain and contextual information. In this work, our main contribution are:

1. Joint end-to-end learning model for event extraction and context identification;
2. We treat the context classification as multiclass classification problem and propose use of *Adaptive Sample Weighting* for end-to-end model learning.

3. Experimental Setup

Dataset We used the provided CMED dataset [7] and it consisted of a training set of 350 clinical notes, a development set of 50 clinical notes and a test set of 100 clinical notes. It is relevant to note that the train and dev set have a similar average length distribution; 977 on train and 947 on dev respectively. The longest clinical note in train + dev has a length of 4265. The test set had average note length of 901 words and longest note contained 2887 words.

Approach Our joint learning model is based on a pipeline (Fig.1) whose architecture of consists of two encoder modules i.e. static embeddings [8] (FastText, Glove and POS-Tagging-scheme) and contextual embedding module (BERT-based), a feature concatenation layer, and two separate linear head layers. The first head is responsible for the medication extraction and event classification, while the second head determines the context information for any detected *Disposition* event cases by the first linear head.

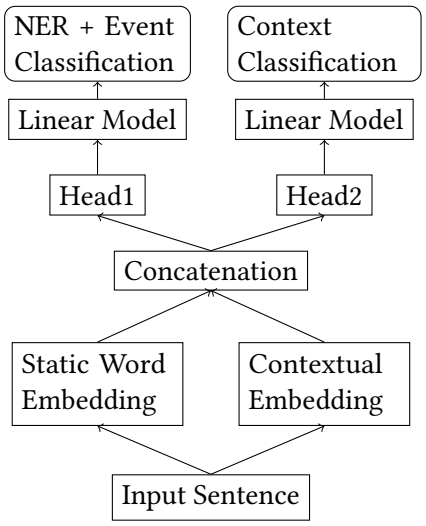

Figure 1: Schematic Pipeline of the CME^2 Net

Prior to main experiments, we performed a model selection via the 1st task using BERT-base-cased, Bio-ClinicalBERT [9], and BioELECTRA [10]. We found Bio-ClinicalBERT model to be the best among the three models and we further used it for rest of our experiments.

Loss with adaptive sample weighting We perform the joint training by using an additive weighted cross-entropy loss function denoted by:

$$\mathbf{L}(\mathscr{L}_1, \mathscr{L}_2) = \mathscr{K}_1\mathscr{L}_1 + \mathscr{K}_2\mathscr{L}_2 \tag{1}$$

Here, $\mathscr{K}_1$ and $\mathscr{K}_2$ indicates the softmax weights of the losses. We use AdamW optimizer for training the model. For medication extraction and event classification is represented by $\mathscr{L}_1$ and that for the context classification task is represented by $\mathscr{L}_2$. Each $\mathscr{L}_j$ corresponds to the weighted cross-entropy loss (CE) which is inspired by [11] and is denoted by:

$$\mathscr{L}_j = \sum_{i=1}^{N}(-y_i \ln(p_i)) * (1- < p_i >)^{0.1} \tag{2}$$

Here, p_i, y_i refers to the prediction probability and labels respectively. The calculated value $(1- < p_i >)^{0.1}$ do not participate in gradient descent as they are detached from the computational graph and are treated as constants. We hypothesize this to be similar to hard negative mining [12] in computer vision, thereby we force the model to learn a good generalization, by putting more focus on the under-confident examples.

Metrics For medication extraction task (task1), Lenient F1 score was used as the primary evaluation metric. For event classification (task2), macro f1 score and for context classification task(task3) F1-score was used.

4. Results

With $\mathscr{K}_2$=0, we perform the following mentioned tests (a) RNN-LSTM refinement [4] for medication extraction task; (b) Effect of word embedding(WE), dropout(DP), class weights(CW) and sentence-truncation(ST); (c) Effect of using bidirectional LSTM layer on word embeddings and upsampling. Using the obtained best model, we run the joint training for the entire model.

Model	Task1	Task2
ClinicalBERT	0.9753	0.8513
+ biLSTM rf. + FT Head1	0.9749	0.8495
+ biLSTM rf. + FT BiLSTM	0.9758	0.8434
+ biLSTM rf. + FT Complete	0.9326	0.8598
+ biRNN rf. + FT Head1	0.9768	0.8524
+ biRNN rf. + FT biRNN	0.9753	0.8482
+ biRNN rf. + FT Complete	0.9742	0.8522

Table 1: Effect of RNN-LSTM refinement (rf.)

4.1. Dev experiments

RNN-LSTM refinement As shown in Table 1, our base model Bio-ClinicalBERT obtained a lenient F1-score of 0.9753 for task 1 and a macro lenient F1-score 0.8513 for task 2. When

Model	Task1	Task2
ClinicalBERT	0.9753	0.8513
+ LSTM + CE(1,0)	0.9769	0.8451
+LSTM + WE + CE(1,0)	0.9690	0.8548
+DP(0.5)+WE+CE(0.5,0.5) [3]	0.9757	0.8150
+DP(0.5) +WE+ CE(0.5,0.5)+UPSAMPLING	0.9758	0.8636
+WE+ CE(0.33,0.67)+UPSAMPLING [4]	0.9762	0.8980

Table 3
Effect of using additional bi-LSTM layer and upsampling on development set

bi-LSTM layer is added to the base model, the F1 score for task 1 slightly increased to 0.9758 but macro F1 for task 2 decreased to 0.8434. Similarly, when bi-RNN layer is added to the base model the F1-score for task 1, remains unchanged but the macro F1-score for task 2 decreased to 0.8482.

Effect of word embedding Next, we tried different combinations of dropout {0.2, 0.5}, adding static word embedding (WE) layer, assigning class weights (CW) and sentence truncation (ST) (refer Table 2). We observed that sentence truncation and using class weights did not improve the results. We then added static word embeddings, for which we used Glove and FastText embeddings trained on Open Access Case Reports [8]. We observed that adding word embeddings and dropout rate of 0.5 increased the F1 score for task 1 to 0.9792 and the macro F1-score for task 2 to 0.8702.

Model	Task1	Task2
ClinicalBERT	0.9753	0.8513
+ DP(0.2)	0.9743	0.8377
+ DP(0.2) + ST	0.9768	0.8233
+ DP(0.2) + CW + ST	0.9739	0.8238
+ DP(0.2) + WE	0.9684	0.8529
+ DP(0.2) + WE + ST	0.9751	0.8355
+ DP(0.2) + WE + CW + ST	0.9723	0.8212
+ DP(0.5) + WE	0.9792	0.8702
+ DP(0.5) + WE + ST	0.9722	0.8443
+ DP(0.5) + WE + CW + ST	0.9724	0.8277

Table 2: Effect of word embedding(WE), dropout(DP), class weights(CW) and sentence-truncation(ST) on development set

Effect of using bidirectional LSTM layer We added additional biLSTM over static word embeddings which gave an F1 score of 0.9769 on task 1 and 0.8451 on task 2 which did not improve the model results. From all the combinations which we tried on top of Bio-ClinicalBERT, the best was using a dropout of 0.5 with additional static word embedding encoder layer. Now, we trained the model with cross entropy loss with ($\mathscr{K}_1$,$\mathscr{K}_2$)=(0.5,0.5) which give F1 score of 0.9757 , 0.8150 and 0.5144 on task 1,2 and 3 respectively. We then perform the model training with upsampling (refer Table 3) and we observe that the it enhances the performance on the three tasks. Post-evaluation, we perform an additional model run with upsampling and ($\mathscr{K}_1$,$\mathscr{K}_2$)=(0.33,0.67) where we notice that model performance is increased as it obtains F1 score of 0.9762, 0.8980 and 0.5857 on the three tasks respectively.

[3]Our submitted system
[4]Post-evaluation system

	Task(s)	Metric	Max	Min	CME^2 Net	Overall Rank
R1	NER	Strict F1	0.9716	0.0913	0.9588	2nd
		Lenient F1	0.9846	0.0945	0.9831	
	NER + Event	Lenient micro F1	0.9225	0.2170	0.9101	3rd
		Lenient macro F1	0.8348	0.2666	0.8186	
	E2E	Combined Lenient F1	0.6647	0.0219	0.6145*	2nd
R2	Event	Lenient micro F1	0.9379	0.4243	0.9272	4th
		Lenient macro F1	0.8673	0.2663	0.8417	
	Event+Context	Combined Lenient F1	0.6766	0.0046	0.6504	2nd
R3	Context	Combined Lenient F1	0.7297	0.0209	0.6912*	2nd

Table 4
Overall Model Performance on test set in comparison all systems submitted at n2c2 2022 Track-1 challenge; *- scores correspond to post-evaluation system result

4.2. Test experiments results

Our best submitted system obtained **0.9831** Lenient F1-score on medication extraction task (task 1) and 0.9588 F1-score for strict matching obtaining overall second position. With the gold labels for task 1, our best submission obtained **0.9272** F1-score on event classification for strict matching to obtain fourth position. Finally, our post-evaluation system obtained **0.6912** F1-score on context classification to obtain overall second position on the leaderboard.

5. Discussion

Error analysis on development set : We observe that our model is able to extract 962 out of the 1010 medication names exactly. The medication instances where the model was not successful included mainly "Insulin NPH" and "contrast dye" which were detected separately. If some medication is agglomerated such as "lipitor20", the model extracts it entirely. Medication brand names such as "CARDURA" , "Lamictal" were also observed to be problematic. Other examples include "Lisinopril/HCTZ", "Ca 600/vit D" where the model detects the medication separately. We also observed cases where certain medication occurred multiple times in the document and the model tagged them differently. This can be attributed the tagging inconsistency problem in NER. Out of 201 cases, our model classified 167 event instances correctly. For the misclassification, we noticed that 2/3rd cases where model confuses between *Undetermined* and *Disposition*/*UnDisposition* classes which can be attributed local context of the medication. Although, we observe that our models misclassifies 70 times out of 167 context dimensions. Further, we notice that out of 70 misclassification[5], model struggles the most with `Action` (43 times), `Certainty` and `Temporality` (17 times), `Actor` (13 times) and the least with `Negation` (4 times). For `Action`, the error can be attributed to the fact that often medication is mentioned as a list or patient history that are the source of error. The narrative of the clinical report often gets confusing with *Start* or *UniqueDose*, similarily for `Certainty`, *Hypothetical* and *Conditional*

[5]Note: Detected medication change has to be annotated for all the five contextual dimensions by the model, a single mis-classification may involve overlap of multiple errors

situation get confusing. In case of `Actor`, the model often confuses as the *Patient*/*Physician* were mentioned before in the text. It is also interesting to note, model was able to predict action in colloquial language usage such as "inc"/ "taper off" whereas gold annotation marked them wrong.

6. Conclusion

In this work, we proposed CME^2 network, a joint learning model for identification and contextual classification for medication change in clinical notes, as part of the n2c2 challenge. For future work , we would like to look into multilabel setting for the medication change context classification to explore the possibility of multiple context annotation and look into methods to incorporate more context information in the clinical note.

References

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, CoRR abs/1706.03762 (2017). URL: http://arxiv.org/abs/1706.03762. arXiv:1706.03762.

[2] S. Sohn, S. P. Murphy, J. J. Masanz, J.-P. A. Kocher, G. K. Savova, Classification of medication status change in clinical narratives, in: AMIA Annual Symposium Proceedings, volume 2010, American Medical Informatics Association, 2010, p. 762.

[3] G. Gkotsis, S. Velupillai, A. Oellrich, H. Dean, M. Liakata, R. Dutta, Don't let notes be misunderstood: A negation detection method for assessing risk of suicide in mental health records, in: Proceedings of the Third Workshop on Computational Linguistics and Clinical Psychology, 2016, pp. 95–105.

[4] L. Rumeng, N. Jagannatha Abhyuday, Y. Hong, A hybrid neural network model for joint prediction of presence and period assertions of medical events in clinical notes, in: AMIA Annual Symposium Proceedings, volume 2017, American Medical Informatics Association, 2017, p. 1149.

[5] P. Bhatia, B. Celikkaya, M. Khalilia, Joint entity extraction and assertion detection for clinical text, arXiv preprint arXiv:1812.05270 (2018).

[6] H. Harkema, J. N. Dowling, T. Thornblade, W. W. Chapman, Context: an algorithm for determining negation, experiencer, and temporal status from clinical reports, Journal of biomedical informatics 42 (2009) 839–851.

[7] D. Mahajan, J. J. Liang, C. Tsou, Toward understanding clinical context of medication change events in clinical narratives, CoRR abs/2011.08835 (2020). URL: https://arxiv.org/abs/2011.08835. arXiv:2011.08835.

[8] Z. N. Flamholz, A. Crane-Droesch, L. H. Ungar, G. E. Weissman, Word embeddings trained on published case reports are lightweight, effective for clinical tasks, and free of protected health information, Journal of Biomedical Informatics 125 (2022) 103971.

[9] K. Huang, J. Altosaar, R. Ranganath, Clinicalbert: Modeling clinical notes and predicting hospital readmission, arXiv preprint arXiv:1904.05342 (2019).

[10] K. Raj Kanakarajan, B. Kundumani, M. Sankarasubbu, Bioelectra: pretrained biomedical

text encoder using discriminators, in: Proceedings of the 20th Workshop on Biomedical Language Processing, 2021, pp. 143–154.

[11] X. Li, X. Sun, Y. Meng, J. Liang, F. Wu, J. Li, Dice loss for data-imbalanced nlp tasks, arXiv preprint arXiv:1911.02855 (2019).

[12] R. B. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, CoRR abs/1311.2524 (2013). URL: http://arxiv.org/abs/1311.2524. arXiv:1311.2524.