A Conceptual Framework for Clinical Note Analysis

Kangyu Zheng, Yanjun Cui, Hao Zou
Department of Computer Science
University of Minnesota
Minneapolis, MN
zhen0196@umn.edu, cui00022@umn.edu, zou00080@umn.edu

Abstract

Clinical notes are text documents that are created by clinicians for each patient encounter [7]. We present three representative models, an attention model and two bidirectional Gated recurrent units (GRU) models, which are able to predict medical results for Intensive Care Unit(ICU) patients according to their clinical notes from clinical text. The multi-layers bidirectional GRU model preforms best. Furthermore, since we are predicting from clinical notes, we can generate explainable mortality prediction model, from which we can get meaningful explanations.

1 Introduction

While Computer Vision has long been regarded as a mature application for medical AI such as medical images analysis, the robust performances of natural language processing(NLP) in this area has attracted researchers attention recently. For instance, using the knowledge map of medical materials, which can be extracted from medical records, to form an optimized system for search recommendation and improves semantic understanding at the meantime. Since almost all medical knowledge of human beings is stored in words, which include medical textbooks, documents and medical records, then in theory, if there is a machine that can understand all words, it can master all human medical knowledge.

Previous work has been worked on using patient's physical sign data, such as heart rate, temperature, white cell blood count in 72-hours to predict the mortality and also using Shapley Value to explain the prediction[3]. However the challenges is although we can know what part of the data contribute most to the prediction, the data itself cannot tell us why the data will change such that it leads to the result. Clinical notes, however, record the patient's admission status, treatment in the ICU and doctor's judgment. It would be very useful for doctor to make further treatment if the model can give explanation based on these human-readable messages. So we decide to make a explainable prediction model based on clinical note.

1.1 Challenges

The challenge of using NLP to analyze electronic version of clinical notes is that there are lots of noises, sparseness and bias for each of patient[11]. Because the disease works on different patients can still cause different effects. Therefore, figuring out how to work on these biased dataset is one of our challenges.

Second, the clinical notes unusually contains many negative, uncertain and conditional statements[9]. These statements will increase the difficulty for the system to understand these information. Also, the clinical notes usually contains the medical history of patients, even their families' medical history. As for doctor, they are able to use their past experience to evaluate the current condition of the patients. But as for a computer system or program, it might be hard to understand these words and make

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

predictions.

In addition, the dataset is imbalanced. Around 90% of dataset has labeled the patient is alive, with 10% of the dataset labeled the patient is died. This causes that we got all predicted results are 0 (alive) at first. The accuracy is 90% but the AUC score is pretty low. They, we over-sampled the dataset to figure out this problem.

1.2 Related work

In recent years, there are a growing number of researches using NLP techniques to study the clinical notes. Some researchers use natural language processing techniques to recognize certain diseases, such as critical limb ischemia (CLI)[1] and youth depression[5]. They used neural networks and some machine learning algorithms such as decision trees to create the classification model. And models they created shows they their power in classification the disease.

Researchers also use NLP to understand the chronic diseases[11] and develop the medical subdomain classifiers based on the clinical notes[12].They used convolutional neural network, bag of words and supervised learning-based NLP to developed the classifier. By using clincal NLP, they are able to classify and group more dataset than traditional ways.

The MIMIC-III clinical notes dataset has been previously used by [8] to identify the procedures and diagnoses. They used long short-term memory (LSTM) to build the neural network. And their models can predict top-10 diagnoses and procedures with 80.3% and 80.5% accuracy, whereas the top-50 diagnosis and procedures are predicted with 70.7% and 63.9% accuracy.

2 Methodology

Figure 1 describes the overview of our methodology pipeline of the research. Our methodology involves: information extraction, data preprocessing, feature extraction and model training and testing. Section 2.1, 2.2 describe each step in more detail.

To achieve our goal, we think of two possible ways. The first one is to use transformer model with self-attention. The second one use multiple layers LSTM/GRU model with attention. Both of these methods are proved to be useful in text classification task, but we do not know which method fit better under clinical text.

We applied an attention mechanism to select the nodes that are relevant in the networks. It pays greater attention to certain factors when processing the data. These attention weights are then applied to the base representation, and the result is passed through an output layer (Mullenbach et al., 2018). We now will describe each part of our model in more detail. To improve the performance of the model pipeline, we also added multi-layer bidirectional GRU to aggregate input information from both previous time steps and later time steps.

2.1 Data Prepossessing

For the data we used Medical Information Mart for Intensive Care III(MIMIC-III) dataset[6]. The MIMIC-III is a large, freely-available database comprising de-identified health-related data associated with over forty thousand patients who stayed in critical care units of the Beth Israel Deaconess Medical Center between 2001 and 2012. This dataset has been widely used in clinical note related research. To extract clinical note from MIMIC-III dataset, we follow the prepossessing method used in ISeeU[3] paper, writing a SQL query to extract the note and category the death and alive according to whether the death time is in the period of ICU in and out time. The total number of clinical notes is 22,055 with labels 1 (death) or 0 (alive). However, the dataset is imbalanced. It contains 1,973 notes is label 1, while others are label 0. Each clinical notes contains patients name, age, sex, history of present illness and some other descriptions of state of illness.

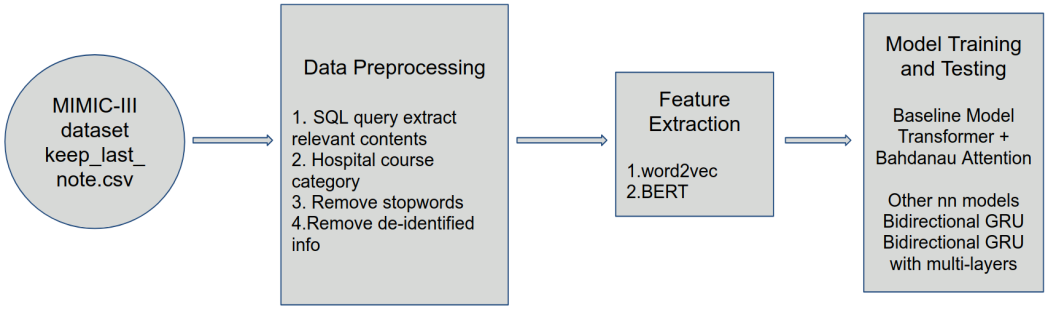

Figure 1: An overview of our methodology pipeline.

In this project, we are focusing on the human readable clinical notes part with the label about whether the patients is alive or not. Therefore, only the free-text clinic note section from the dataset was used. Specifically, the focus was on *hospital course* category as it contains direct and crucial information compared to other categories. The data was preprocessed to produce clean dataset for training. Stop words, ICD9 labels and de-identified information such as words inside brackets were removed. *word2vec* was used for feature extraction.

2.2 Model Training and Testing

In our study, one baseline approach was first created: Transformer and Attention mechanism. Specifically, we were using Bahdanau attention [2] in the model. Then we implemented three other deep neural network models to improve the performance of the baseline model.

2.2.1 Baseline Model

transformer + Bahdanau Attention: Our first baseline model is transformer framework plus Bahdanau mechanism.
Transformers: The basic mechanism we used in our model is Transformers. We used word2vec to transform document to corresponding vectors. We also tried BERT to improve the model's performance, however, it showed low AUC value.

Encoding: At the base layer of the model, we have pre-trained embeddings for each word in the document. Then Gated Recurrent Units were applied in encoding part. LSTM is a special type of RNN. It is a model that applies the results of past learning to the current learning, but this general RNN has many drawbacks. For example, if we want to predict the last word of "the dogs like the bones" because it is only predicted in the context of this sentence, it will be easy to predict that the word is "bones". In such a scenario, the interval between the relevant information and the predicted word position is very small, and the RNN can learn to use the previous information.

One of the reasons for using LSTM is to solve the problem that the Gradient Error of RNN Deep Network accumulates too much, so that Gradient returns to zero or becomes infinite, so the optimization cannot be continued. While the structure of GRU is simpler: one less gate than LSTM, so there are fewer matrix multiplications. In the case of large training data, GRU can save a lot of time. GRU is very similar to LSTM. Compared with LSTM, GRU removes the cell state and uses the hidden state to transmit information. It contains only two doors: update door and reset door. Figure 2 indicates a typical structure of GRU where Z is the update gate and r is the reset gate.

Decoding: Bahdanau attention was applied in decoding part. In the traditional seq2seq model, the encoder encodes the input sequence into a context vector, and the decoder uses the context vector as the initial hidden state to generate the target sequence. As the length of the input sequence increases, it is difficult for the encoder to encode all input information into a single context vector, then the coding information is missing, and it is difficult to complete high-quality decoding.

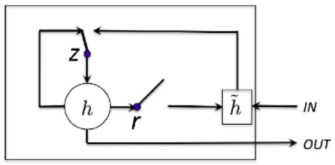

Figure 2: Typical structure of a GRU.[10]

The attention mechanism is to calculate the attention of the hidden state at each position based on the hidden state, input or output of the decoder at each moment of decoding and weighted to generate the context vector for the current moment decoding. The attention mechanism is introduced to improve the accuracy of prediction.

Bahdanau Attention came from a paper by Dzmitry Bahdanau. The paper aimed to improve the sequence-to-sequence model in machine translation by aligning the decoder with the relevant input sentences and implementing Attention [2]. It can align the hidden state of the decoder with the output of all positions of the encoder through linear combination to obtain the context vector, which is used to improve the sequence-to-sequence model. The encoder generates a hidden state vector for each input vector, then use the hidden state at the previous moment and the encoder's output for each position to calculate the alignment score (calculated using a feedforward neural network). The final moment hidden state of the encoder can be used as the initial moment hidden state of the decoder. The alignment score of the hidden state output by the decoder at each position of the encoder at the previous moment is then transformed into a probability distribution vector through softmax function. According to the probability distribution of the alignment score, the output of each position of the weighted encoder is obtained as well as the context vector. At last, splice the context vector and the corresponding embedding output of the encoder at the previous moment, as the encoder input at the current moment, and generate new output and hidden state through the RNN network.

A fully connected neural network then was used to get our final result.

$$\mathrm{s}_t = f(s_{t-1}, c_t, y_{t-1})$$

$$\alpha(s_{t-1}, X) = softmax(tanh(s_{t-1}W_decoder + XW_encoder)W_alignment), c_t = \sum_i \alpha_{ti} x_i$$

2.2.2 Other Neural Network Models

Three deep neural network models were implemented to improve the performance of the baseline model. We only applied Bahdanau Attention in single layer GRU in our baseline model. We then try GRU with multiple layers and bidirectional mechanism as well to see if the result can get improved.

Bidirectional GRU: Bidirectional GRU is a type of bidirectional recurrent neural networks. It only has the input and forget gates and allows for the use of information from both precious and later time steps. It can capture long-term dependencies to retain useful information. This kind of framework can adjust important needs in Natural Language Process problems. Besides, in our study, we found that Bidirectional GRU with multiple layers can achieve better performances compared to single-layer GRU. Bidirectional GRU can connect two hidden layers of opposite directions to the same output, which has been shown being able to improve the basic performance. This mechanisms make the network receivr information from previous and later states simultaneously, which is especially useful when the later context of the input is needed [4]. Furthermore, to achieve even greater results , multiple RNN(GRU) framework was implemented. We added Dropout and BatchNormalization layer as well, and two bidirectional GRU were on the top of them. We then compare the results between Attention mechanism, Bidirectional framework and Multiple RNN system.

3 Results

There are three representative models, the first one is Bahdanau attention model, the second and third one are single layer and multi-layers bidirectional GRU models.

3.1 Bahdanau attention

Here is our current results. By using word2vec transformer with attention models, we finally obtained the following results in figure 1. Yellow color means that has higher attentions. Finally, after we finished all training process and do the prediction on test set, when we apply the model to the training dataset, we receive an AUC score is 0.85, with 0.82 accuracy, which shows in figure 2. However, when we use the model to do prediction on our test dataset, which shows in figure 3. The AUC score is 0.73, but the accuracy is pretty low (0.48).

Figure 3: Attention weight graph after training with word2vec.

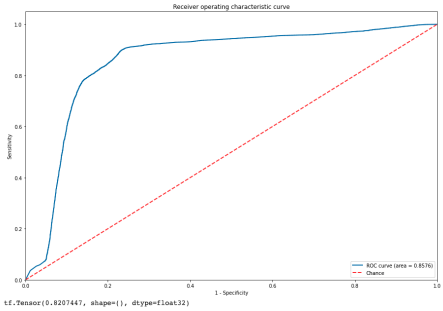

Figure 4: Plot for ROC curve with word2vec transformer.

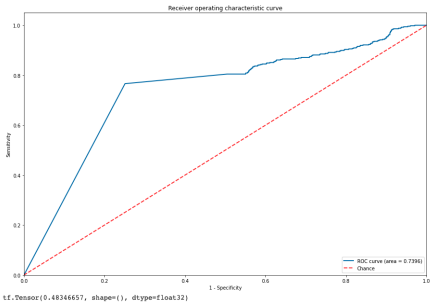

Figure 5: Plot for ROC curve with word2vec transformer for test set.

3.2 Bidirectional GRU

By using single-layer bidirectional GRU, the performance of training dataset (figure 4) is not bad, with AUC score and accuracy are 0.9175 and 0.840 respectively. However, in test dataset, which shows in figure 5, the AUC score is pretty low (only 0.59).

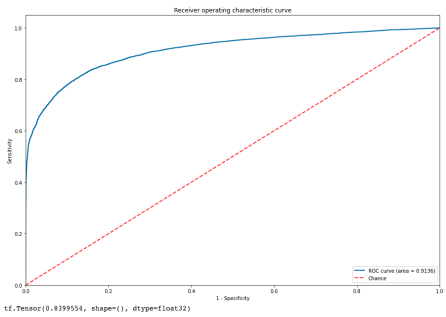

Figure 6: Plot for ROC curve with single layer bidirectional GRU for training set.

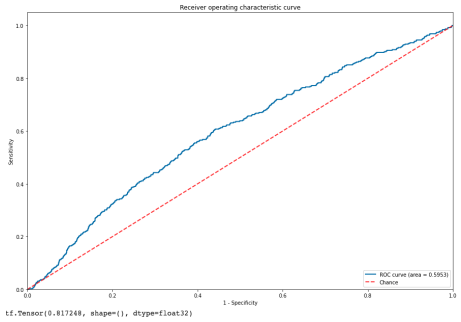

Figure 7: Plot for ROC curve with single layer bidirectional GRU for test set.

As for multi-layers bidirectional GRU with batch normalization and drop out, we obtain higher scores in accuracy and AUC.For training dataset (figure 6), the ROC curve is 0.95 while the accuracy is 0.89. Figure 7 shows the test dataset results, the score is lower than than training dataset, with AUC score is 0.71 and accuracy is 0.85. However, this results is much better than Attention models and single layer bidirectional GRU.

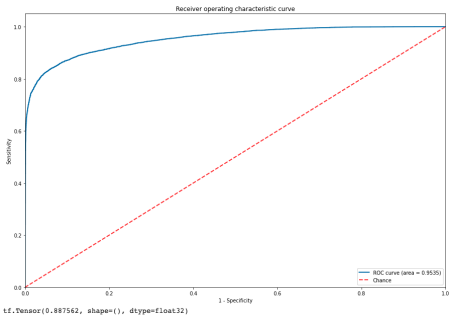

Figure 8: Plot for ROC curve with multi-layers bidirectional GRU for training set.

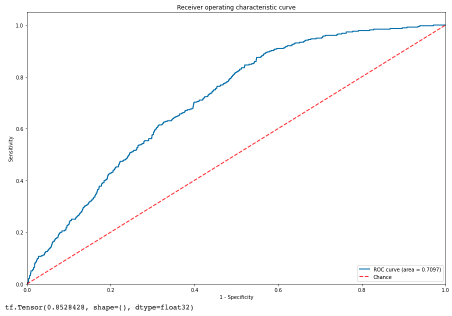

Figure 9: Plot for ROC curve with multi-layers bidirectional GRU for test set.

4 Discussion

In this project, we found out some problems that prevent us to get higher accuracy and AUC.
The first one is the data we chosen. We choose to analyse clinical notes in this project. However, those clinical notes are written in professional way and since none of us has medical background, it is very hard for us to determine which part of the clinical note should be analysed. Also the data itself is very unbalance. Only ten percent of our total dataset is marked as mortality. Although we apply Synthetic Minority Oversampling Technique (SMOTE) to help us improve the imbalance situation, it does not improve the overall prediction to our perspectives.

The second one is the way we process the data. In this project, we only apply NLTK tokenization to tokenize our text. This tokenization only separate words according to white space and remove all punctuation. However, such simple procession does not suit well in healthcare area. For example the phase "heart attack", it will be separated as two independent words "heart" and "attack" and apparently we want our model to treat this as single phases instead of two separated words.

According to the problems mentioned above, we propose that we could use knowledge graph in the future to improve our project. The knowledge graph represents a collection of interlinked descriptions of entities – objects, events or concepts. Knowledge graphs put data in context via linking and semantic metadata and this way provide a framework for data integration, unification, analytics and sharing. We think we could put medical and healthcare term into such knowledge graph so that we can preserve these terms when processing the data and make the prediction more accurate.

5 Conclusion

In this project, we use single-layer recurrent neural network with attention and bidirectional recurrent neural network to analyse clinical note data from MIMIC-III dataset. Our result shows that multi-layer bidirectional recurrent neural network model preforms better in training and testing phase according to AUC measurement. At last we review our problems encountered during the project and propose one potential solution.

References

[1] Naveed Afzal, Vishnu Priya Mallipeddi, Sunghwan Sohn, Hongfang Liu, Rajeev Chaudhry, Christopher G. Scott, Iftikhar J. Kullo, and Adelaide M. Arruda-Olson. Natural language processing of clinical notes for identification of critical limb ischemia. *International Journal of Medical Informatics*, 111:83 – 89, 2018.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2014. cite arxiv:1409.0473Comment: Accepted at ICLR 2015 as oral presentation.

[3] William Caicedo-Torres and Jairo Gutierrez. Iseeu: Visually interpretable deep learning for mortality prediction inside the icu. *Journal of Biomedical Informatics*, 98:103269, Oct 2019.

[4] Cesar; Wicent Sy Luke Huang, Jinmiao; Osorio. An empirical evaluation of deep learning for icd-9 code assignment using mimic-iii clinical notes. 177, 2019.

[5] Geraci J, Wilansky P, de Luca V, Roy A, Kennedy JL, and Strauss J. Applying deep neural networks to unstructured text notes in electronic medical records for phenotyping youth depression. *Evid Based Ment Health*, 2020.

[6] Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3:160035, 2016.

[7] James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. Explainable prediction of medical codes from clinical text, 2018.

[8] Siddhartha Nuthakki, Sunil Neela, Judy W. Gichoya, and Saptarshi Purkayastha. Natural language processing of mimic-iii clinical notes for identifying diagnosis and procedures with neural networks, 2019.

[9] Sujan Perera, Amit Sheth, Krishnaprasad Thirunarayan, Suhas Nair, and Neil Shah. Challenges in understanding clinical notes: Why nlp engines fall short and where background knowledge can help. 10 2013.

[10] Rajib Rana. Gated recurrent unit (gru) for emotion classification from noisy speech. 2016.

[11] Sheikhalishahi S, Dudley JT Miotto R, Lavelli A, Rinaldi F, and Osmani V. Natural language processing of clinical notes on chronic diseases: Systematic review. *JMIR Med Inform*, 2019.

[12] Wei-Hung Weng, Kavishwar B. Wagholikar, Alexa T. McCray, Peter Szolovits, and Henry C. Chueh. Medical subdomain classification of clinical notes using a machine learning-based natural language processing approach. *BMC Medical Informatics and Decision Making*, 17, 2017.