IITP at MEDIQA 2019: Systems Report for Natural Language Inference, Question Entailment and Question Answering

Dibyanayan Bandyopadhyay[1] **Baban Gain**[1] **Tanik Saikh**[2] **Asif Ekbal**[2]
Government College Of Engineering And Textile Technology, Berhampore[1]
Indian Institute of Technology Patna[2]
{dibyanayan,gainbaban}@gmail.com[1]
{1821cs08, asif}@iitp.ac.in[2]

Abstract

This paper presents the experiments accomplished as a part of our participation in the MEDIQA challenge, an (Abacha et al., 2019) shared task. We participated in all the three tasks defined in this particular shared task. The tasks are *viz. i. Natural Language Inference (NLI) ii. Recognizing Question Entailment(RQE)* and their application in medical *Question Answering (QA)*. We submitted runs using multiple deep learning based systems (runs) for each of these three tasks. We submitted five system results in each of the NLI and RQE tasks, and four system results for the QA task. The systems yield encouraging results in all the three tasks. The highest performance obtained in NLI, RQE and QA tasks are 81.8%, 53.2%, and 71.7%, respectively.

1 Introduction

Natural Language Processing (NLP) in biomedical domain is an essential and challenging task. With the availability of the data in electronic form it is possible to apply Artificial intelligence (AI), machine learning and deep learning technologies to build data driven automated tools. These automated tools will be helpful in the field of medical science. An ACL-BioNLP 2019 shared task, namely the MEDIQA challenge aims to attract further research efforts in NLI, RQE and their application in QA in medical domain. The motivation of this shared task is in a need to develop relevant methods, techniques, and gold standard data for inference and recognizing question entailment in medical domain and their application to improve domain specific Information Retrieval (IR) and Question Answering (QA) systems. The MEDIQA has defined several tasks related to *Natural Language Inference, Question Entailment and Question Answering* in medical domain. We participated in all the three tasks defined in this shared task. We offer multiple systems for each the tasks. The workshop comprises of three tasks namely *viz. i. Natural Language Inference (NLI): This task involves in identifying three inference relations between two sentences: i.e. Entailment, Neutral and Contradiction (Romanov and Shivade, 2018) ii. Recognizing Question Entailment (RQE): This task focuses on identifying entailment relation between two questions in the context of QA. The definition of question entailment is as follows: "a question A entails a question B if every answer to B is also a complete and/or partial answer to A" (Abacha and Demner-Fushman, 2019) and iii. Question Answering (QA): The goal of this task is to filter and improve the ranking of automatically retrieved answers. The existing medical QA system namely CHiQA is applied to generate the input ranks. (Harabagiu and Hickl, 2006; Abacha et al., 2017; Abacha and Demner-Fushman, 2019)*. We participated in all the three tasks defined above and submitted the results. Our proposed systems produce encouraging results.

2 Proposed Method

We propose multiple runs for each of the three tasks. The following subsections will discuss the methods applied to tackle each of these tasks.

2.1 Natural Language Inference

In the task 1 the system has to decide the entailment relationship between a pair of texts i.e. either they are *Entailment, Contradiction or Neutral*. The input to this task are sentence pairs and as output we wish to get the entailment relation between those two pieces of texts. We propose five runs for this task. The following set of hyper parameters are applied for the following runs. **Batch**

Proceedings of the BioNLP 2019 workshop, pages 517–522
Florence, Italy, August 1, 2019.

Size = 32, **Learning Rate** = 2e-5, **Maximum Sequence Length** = 128, **number of epochs** = 10. The following points will discuss the approaches (i.e. runs).

Run 1: Our first proposed method is based on a BioBERT (Lee et al., 2019) model, i.e. a Bidirectional Encoder Representation from Transformer model pre-trained on biological data (both on Pub Med abstracts and PMC full-text articles). After getting the vector corresponding to the special classification token ([CLS]) from final hidden layer of this mode, we use it for classification. We use 2 dense (feed forward) layers and a softmax activation function at the end. Only the feed forward part is trained end to end for 10 epochs after getting output vector from BioBERT. In this method no fine tuning is used. This method yields an accuracy of 60.8%.

Run 2: The second approach is based on the Bidirectional Encoder Representation from Transformer (BERT) model (bert-base-uncased) (Devlin et al., 2018). We make use of this to get the embedding of the inputs to this system. Instead of using 2 feed forward layers at the end, we choose to use only 1 feed forward layer. The full model is then trained in an end to end manner. All of the parameters of BERT and the last feed forward layer are fine-tuned jointly for 10 epochs to maximize the log-probability of the correct label (Entailment, Neutral or Contradiction). This model produces an accuracy of 71.7%.

Run 3: We use a BioBERT model for this system. This BioBERT model is pre-trained on PubMed abstracts only. We apply only one feed forward layer at the end. The full model is fine tuned as described for the system in run 2. This method gives output with an accuracy of 77.1%.

Run 4: The system proposed in this run is same as the model of Run 3. The differences between them are as follows:

- The BioBERT model we used is a pre-trained model on both the PubMed abstracts and PMC full-text articles instead of only PubMed abstracts as in case of in run 3.
- Here, we combine the full dataset of MedNLI (14049 sentence pairs) for training. Whereas in the previous run we made use only 11232 sentence pairs for training.

Following these changes in run 4, the accuracy increases from 77.1% in run 3 to 80.3% in run 4.

Run 5: This model is the combination of three BioBERT models. Two of them are pre-trained on both the PubMed abstracts and PMC full text articles and the third one is pre-trained only on PubMed abstracts. We fine tune each of the models following the fine tuning process of run 4. We ensemble their predictions by voting each of them for a sentence pair. The label which gets the most vote is selected for final prediction. The accuracy increases to 81.8%.

2.2 Recognizing Question Entailment (RQE)

Recognizing Question Entailment is an important task. The objective of this task is to identify entailment between the two questions in the context of Question-Answering(QA). We use the following definition of question entailment: a question A entails a question B if every answer to B is also a complete or partial answer to A. We make use of the dataset provided by the task organizers'. We submit five runs which are broadly based on two approaches. The approaches are as follows:

- One is based on Siamese architecture (Mueller and Thyagarajan, 2016). This Siamese is based on the recurrent architectures for learning sentence similarity (Mueller et al., 2016). Here we feed the two questions (inputs) to two Bidirectional Long Short Term Memory (Bi-LSTM) (Hochreiter and Schmidhuber, 1997), respectively. Both of their weights are initialized to the same. After obtaining the last hidden representations from both of these Bi-LSTMs, we concatenate them. This vector represents our input sentence pair. We feed this vector to a feed forward neural network layer. At the end, there is a softmax layer to perform a 2-class classification (to Yes/No).

- In another one, we train (fine-tuned) a BioBERT model as described in the NLI task. We used the BioBERT model to perform a 3-way classification of a sentence pair into entailment, neutral or contradiction. The same approach is used here in RQE to classify a pair of questions into Yes or No. The hyper-parameters used in fine-tuning the BioBERT model are same as of task 1 (NLI) except here the training iteration is (i.e. epoch) 5. This is done so because the loss is decreasing rapidly

between two training epochs, indicating over fitting on the train set.

Our proposed approaches are based on these methods with slight variations. The following points will show them.

Run 1: Each question pair is having two questions (namely chq and faq). We assume the first question as the premise and the second one as the hypothesis. We extract these two questions from the training set. We obtain the vector representation of each word using Gensim Word2Vec (Řehůřek and Sojka, 2010). The vector size is 50. Then vector representations of the words for both the question are fed to Siamese Network of Bidirectional LSTMs. We train the model with 50 epoch and achieve 53.2% accuracy in test set.

Run 2: In this method we make use of the BioBERT model. The model is pre-trained on PubMed abstracts and PMC full text articles. This task is essentially a sentence pair classification task. For each sentence pair we obtain a vector corresponding to ([CLS]) token at the last layer. The vector is subsequently fed into two dense layers followed by a final layer having softmax activation function layer. No fine-tuning is used here. We obtain an accuracy of 50.6%.

Run 3: Here also we use BioBERT, but it is pre-trained on PubMed abstracts only. We fine tune the model on the RQE training set consisting of 8588 pairs. A feed forward layer with final layer with softmax activation is used at the end for 2-way classification. We obtain an accuracy of 48.1%.

Run 4: Instead of training word vector representation from scratch using Gensim Word2Vec, as in run 1, we obtain the vector representations of words from a trained Google News corpus (3 billion running words) word vector model (3 million 300-dimension English word vectors). The architecture is same as what is there in the run 1. We obtain an accuracy of 50.2%.

Run 5: We use a BioBERT model pre-trained on both PubMed abstracts and PMC full text articles. Then we fine tune on the RQE train set. Everything else is same as in run 3. The accuracy decreases to 48.9%.

2.3 Question Answering (QA)

The objective of this Question Answering task is to filter and improve the ranking of automatically retrieved answers. The input ranks are generated by the existing medical QA system *CHiQA*. We use BERT to predict the reference score between pairs and BM25 (Robertson et al., 2009) to rank between them. First of all, the BERT is used as a sentence pair classifier model. The first token of every sequence is always the special classification token ([CLS]). The final hidden state (i.e., output of Transformer) corresponding to this token is used as the aggregate sequence representation for classification tasks. This final hidden state is a 768 dimensional vector (for bert-base) representing the input sentence pair. This vector is fed subsequently into one or more feed-forward layers with softmax activation function layer. We fine tune the whole system for 10 epochs to predict the reference score of test dataset. The predicted values of these reference score are subsequently used in the BM25 model. *All the hyper parameters setting are same as in Task-1 except here a batch size of 28 is used because the maximum length of sequence is increased from 128 to 256.* It is to be noted that, it is too much memory consuming to train a BERT model with a batch size of 32 and a maximum sequence length of 256. We propose four runs to combat this problem.

Pre-processing: The training dataset is divided into two files (QA-TrainingSet1-LiveQAMed and TrainingSet2-Alexa) both are containing 104 questions. They had 8.80 and 8.34 answers for every question on average, respectively. For a question with *N* answers are converted to *N* pairs with each pair containing the question, one of the answers, and their reference score.

Run 1: All reference scores from training dataset are replaced by 1 if it is 3 or 4, and with 0 otherwise. The range of reference score (as given) is between 1 to 4 in dataset. Here we fine tune a BERT model (bert-base-uncased) with a feed forward layer at the end for 10 epochs to classify a sentence pair into 2 labels (0 or 1). The trained model is then used to predict reference score of test set. From the predicted result obtained from BERT, BM25 score for every question and their corresponding answers are calculated. All answers for a question whose predicted labels are 1 ('YES') are sorted in decreasing order of their BM25 scores. After that all the 'YES' labels are retrieved, and the same procedure is applied for all answers for the same question whose predicted

label is 0 (NO). The obtained accuracy is 57.3%.

Run 2: All reference scores are kept intact, i.e. between 1 to 4. Here we use the BERT model (bert-base-uncased) and fine tune it on train set with a feed forward net at the end. From the predicted result obtained from BERT, pairs whose reference score is 4 or 3 are marked as 'YES' and whose reference score are 2 or 1 are marked as 'NO'. BM25 score for every question and their corresponding answer is calculated. All answers for a question whose predicted label is 4 are sorted by decreasing order of their BM25 score. Same procedure is applied for all answers for the same question whose predicted label is 3,2 and 1, respectively. We obtain an accuracy of 65.1% in this run.

Run 3: Here the validation dataset is also included to the training set. We merged them. Instead of using a BERT model here we use BioBERT model which is pre-trained on PubMed abstracts and PMC full text articles. We fine tune this model as explained in run 1. The rest of the procedures are same as in run 2. The accuracy increases to 67.8%.

Run 4: This method is an ensemble of 5 BioBERT (PubMed-PMC) models and fine tuned on the train dataset. Each of the models is then evaluated on the validation set (which is included in training set of Run 3). It is seen that one of those models performs well than the ensemble of 5 models. The model is then used to predict reference score of the test set. The rest of the procedures is same as what is there in the Run 3. The accuracy is 71.7% for this run.

3 Experiments, Results and Discussions

We submitted system results (runs) for all the three tasks. In all these tasks, we make use of the dataset released as a part of this shared task. In the following we discuss the dataset, evaluation results and the necessary analysis of the results obtained.

Data: In the NLI task, the training and test instances are having 14049 and 405 number of sentence pairs, respectively. In task 2 (i.e. RQE), the training set is having 8588 number of pairs, out of which 4655 and 3933 pairs are having *True* and *False* class, respectively. The validation and test set are having 302 (true: 129 and false: 173) and 230 (true: 115 and false: 115) number of instances. In the QA task, training sets are provided from two domains *viz. LiveQAMed and ii. Alexa*, each having 104 number of questions and at an average of 8.80 and 8.34 number of answers per question. There are 25 number of questions and at an average of 10.44 answers per question are there in the validation set. The test set for this task is having 150 question pairs and on an average 8.5 answer per question.

Runs	Result(Accuracy(%))
1	60.8
2	71.7
3	77.1
4	80.3
5	**81.8**

Table 1: Submission results of all the five runs for the NLI task (Task-1)

Task 1(NLI): In the first task, we propose five runs. In all the tasks, we make use of either BERT or BioBERT models. We merge the input sentences pairs into a single sequence having maximum length of 128. They are separated by a special token ([SEP]). The first token of every sequence is always a special classification token ([CLS]). The final hidden state (i.e., output of Transformer) corresponding to this token is used as the aggregate sequence representation for the classification tasks. This final hidden state is a 768 dimensional vector (for bert-base) representing the input sentence pair. This vector is fed subsequently into one or more feed-forward layers with soft-max activation at the end for 3-way classification (Entailment, Neutral or Contradiction). The results for this task are shown in the Table 1. We have discussed the way we can use a BERT model to perform sentence classification in medical domain. It is observed that an absolute improvement of 5.4% in accuracy has been achieved by using a BioBERT (pre-trained on PubMed abstracts) model in run 3 instead of using the original BERT-base-uncased model (Pre-trained on Wikipedia and Book Corpus (as used in run 2)). The increase in result may be the effect of BioBERT, because the other experimental set up remain same. The reason for using 1 feed forward layer at the end of BERT models in all the runs except the run 1 (no fine tuning), using only one feed forward layer was putting the model into an under fitting state. While in case of fine tuning a large model, one feed forward is enough

Runs	Result(Accuracy(%))
1	**53.2**
2	50.6
3	48.1
4	50.2
5	48.9

Table 2: Submission results in all the five runs for the RQE Task (Task-2)

as suggested by (Devlin et al., 2018). Up to run 3, we make use of 11232 sentence pairs for the training. Those sentence pairs are same as the one used to train several models used in (Romanov and Shivade, 2018). We use the remaining 2817 sentence pairs for validation. The validation set accuracy is always around 3-4% higher than the test case accuracy for all the runs up to run 3. For getting the higher accuracy we combine all the 14049 pairs in the subsequent run. We get the accuracy of 81.8 % which is the highest among all the proposed methods. As per our knowledge, in the official results of NLI task we stand at 12th position among the 17 official teams which participated for the NLI task.

Task 2 (RQE): In the second task i.e. task of Recognising Question Entailment, we propose five runs. The results are shown in the Table 2. It is interesting to note the variation in accuracy for the different runs. Siamese architecture performs much better here. Another peculiarity is that fine tuning BERT hurts the performance while using pre-trained BERT embedding without fine tuning seems to be more useful. This is concluded by observing the results of run 2 and run 3. In run 2, we used only pre-trained BERT embedding for ([CLS]) token for classification, whereas in run 3, we fine tuned the BERT model. The highest accuracy is achieved by a Siamese Model consisting of 2 Bi-LSTMs with shared weights and a dense layers. In this task, 12 teams submitted their systems, and we stood the 10th position.

Task 3 (QA): In this task, we offer 4 runs to tackle the problem. The results for this are shown in the Table 3. As we can see from the above discussions, the systems we build for this task comprises of two components, they are BERT and BM25. The BERT is used to predict the reference score of the test dataset. We rank the

	Results			
Runs	**Accuracy(%)**	**Spearmans Rho**	**MRR**	**Precision**
1	57.3	0.053	0.8241	0.5610
2	65.1	0.042	0.7811	0.7235
3	67.8	0.034	0.8366	0.7421
4	**71.7**	0.024	0.8611	0.7936

Table 3: Results obtained in all the four runs for the QA Task (Task - 3), where, MRR: Mean Reciprocal Rank

predicted scores using BM25. The BM25 part of the system is same for all the runs. In this task, participants are encouraged to compute the Mean Reciprocal Rank (MRR), Precision, and Spearman's Rank Correlation Coefficient as the evaluation measures in addition to Accuracy. We actually used BioBERT instead of original BERT from the run 3, which increases the accuracy with an absolute margin of 2.7% (65.1 to 67.8%). Using BioBERT we observe an improvement in MRR by 5.5%. Our best run with an accuracy of 71.7% attains the position of 6th among 10 teams in the official result.

4 Conclusion and Future Work

In this paper, we present our system details and the results of various runs that reported as a part of our participation in the MEDIQA challenge. In this shared task three tasks, namely *viz. i. Natural Language Inference ii. Question Entailment and iii. Question Answering* were introduced in the medical domain. We offer multiple systems (runs) for each of these tasks. Most of the proposed models are based on BERT/Bio-BERT embedding and BM25. These models yields encouraging performance in all the tasks.In future we would like to extend our work as follows:

- Detailed analysis of the top-scoring models to understand their techniques and findings.
- We can do the task of NLI by fostering an *Embedding from Language model (EMLo)* based model and do a comparative analysis with BERT based model.

References

Asma Ben Abacha, Eugene Agichtein, Yuval Pinter, and Dina Demner-Fushman. 2017. Overview of the medical question answering task at TREC 2017 LiveQA. In *Proceedings of The Twenty-Sixth Text REtrieval Conference, TREC*, pages 15–17.

Asma Ben Abacha and Dina Demner-Fushman. 2019. A question-entailment approach to question answering. *arXiv preprint arXiv:1901.08079*.

Asma Ben Abacha, Chaitanya Shivade, and Dina Demner-Fushman. 2019. Overview of the MediQA 2019 shared task on textual inference, question entailment and question answering. In *Proceedings of the BioNLP 2019 workshop, Florence, Italy, August 1, 2019*. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Sanda Harabagiu and Andrew Hickl. 2006. Methods for using textual entailment in open-domain question answering. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 905–912, Sydney, Australia. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2019. BioBERT: pre-trained biomedical language representation model for biomedical text mining. *arXiv preprint arXiv:1901.08746*.

Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Alexey Romanov and Chaitanya Shivade. 2018. Lessons from natural language inference in the clinical domain. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1586–1596, Brussels, Belgium. Association for Computational Linguistics.