Dr.Quad at MEDIQA 2019: Towards Textual Inference and Question Entailment using contextualized representations

Vinayshekhar Bannihatti Kumar [*] **Ashwin Srinivasan**[*] **Aditi Chaudhary**[*]
James Route **Teruko Mitamura** **Eric Nyberg**
$\{vbkumar, ashwinsr, aschaudh, jroute, teruko, ehn\}$@cs.cmu.edu
Language Technologies Institute
Carnegie Mellon University

Abstract

This paper presents the submissions by Team Dr.Quad to the ACL-BioNLP 2019 shared task on Textual Inference and Question Entailment in the Medical Domain. Our system is based on the prior work Liu et al. (2019) which uses a multi-task objective function for textual entailment. In this work, we explore different strategies for generalizing state-of-the-art language understanding models to the specialized medical domain. Our results on the shared task demonstrate that incorporating domain knowledge through data augmentation is a powerful strategy for addressing challenges posed by specialized domains such as medicine.

1 Introduction

The ACL-BioNLP 2019 (Ben Abacha et al., 2019) shared task focuses on improving the following three tasks for medical domain: 1) Natural Language Inference (NLI) 2) Recognizing Question Entailment (RQE) and 3) Question-Answering reranking system. Our team has made submissions to all the three tasks. We note that in this work we focus more on the task 1 and task 2 as improvements in these two tasks reflect directly on the task 3. However, as per the shared task guidelines, we do submit one model for the task 3 to complete our submission.

Our approach for both task 1 and task 2 is based on the state-of-the-art natural language understanding model MT-DNN (Liu et al., 2019), which combines the strength of multi-task learning (MTL) and language model pre-training. MTL in deep networks has shown performance gains when related tasks are trained together resulting in better generalization to new domains (Ruder, 2017). Recent works such as BERT (Devlin et al., 2018), ELMO (Peters et al., 2018) have shown the efficacy of learning universal language representations in providing a decent warm start to a task-specific model, by leveraging large amounts of unlabeled data. MT-DNN uses BERT as the encoder and uses MTL to fine-tune the multiple task-specific layers. This model has obtained state-of-the-art results on several natural language understanding tasks such as SNLI (Bowman et al., 2015), SciTail (Khot et al., 2018) and hence forms the basis of our approach. For the task 3, we use a simple model to combine the task 1 and task 2 models as shown in §2.5.

As discussed above, state-of-the-art models using deep neural networks have shown significant performance gains across various natural language processing (NLP) tasks. However, their generalization to specialized domains such as the medical domain still remains a challenge. Romanov and Shivade (2018) introduce a new dataset MedNLI, a natural language inference dataset for the medical domain and show the importance of incorporating domain-specific resources. Inspired by their observations, we explore several techniques of augmenting domain-specific features with the state-of-the-art methods. We hope that the deep neural networks will help the model learn about the task itself and the domain-specific features will assist the model in tacking the issues associated with such specialized domains. For instance, the medical domain has a distinct sublanguage (Friedman et al., 2002) and it presents challenges such as abbreviations, inconsistent spellings, relationship between drugs, diseases, symptoms.

Our resulting models perform fairly on the unseen test data of the ACL-MediQA shared task. On Task 1, our best model achieves +14.1 gain above the baseline. On Task 2, our five-model ensemble achieved +12.6 gain over the baseline and for Task 3 our model achieves a a +4.9 gain.

[*] equal contribution

Proceedings of the BioNLP 2019 workshop, pages 453–461
Florence, Italy, August 1, 2019.

2 Approach

In this section, we first present our base model MT-DNN (Liu et al., 2019) which we use for both Task 1 and Task 2 followed by a discussion on the different approaches taken for natural language inference (NLI) (§2.3), recognizing question entailment (RQE) (§2.4) and question answer (QA) (§2.5).

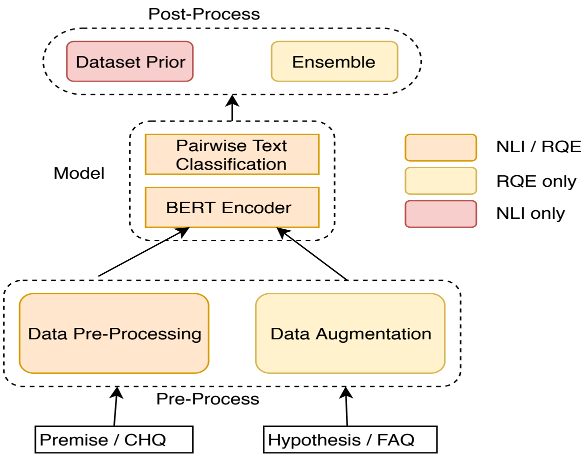

Figure 1: System overview for NLI and RQE task

.

2.1 Task 1 and Task 2 Formulation

Formally, we define the problem of textual entailment as a multi-class classification task. Given two sentences **a** $=a_1, a_2..., a_n$ and **b** $=b_1, b_2, ..., b_m$, the task is to predict the correct label. For NLI, **a** refers to the *Premise* and **b** refers to the *Hypothesis* and the label set comprises of *entailment, neutral, contradiction*. For RQE, **a** refers to the *CHQ* and **b** refers to the *FAQ* and the label set comprises of *True, False*.

2.2 Model Architecture

A brief depiction of our system is shown in Figure 1. We represent components which were used for both NLI and RQE in Orange. An example of this is the Data Pre-processing component. The RQE only components are shown in yellow (eg. Data Augmentation). The components which were used only for the NLI modules are shown in Pink (eg. Dataset Prior). We base our model on the state-of-the-art natural language understanding model MT-DNN (Liu et al., 2019). MT-DNN is a hierarchical neural network model which combines the advantages of both multi-task learning and pre-trained language models. Below we describe the different components in detail.

	Train	Validation	Test
Entailment	3744	465	474
Contradiction	3744	465	474
Neutral	3744	465	474

Table 1: The number of train and test instances in each of the categories of the NLI dataset.

Encoder: Following BERT (Devlin et al., 2018), each sentence pair is separated by a [SEP] token. It is then passed through a lexicon encoder which represents each token as a continuous representation of the word, segment and positional embeddings. A multi-layer bi-directional transformer encoder (Vaswani et al., 2017) transforms the input token representations into the contextual embedding vectors. This encoder is then shared across multiple tasks.

Decoder: We use the *Pairwise text classification output* layer (Liu et al., 2019) as our decoder. Given a sentence pair (**a**,**b**), the above encoder first encodes them into **u** and **v** respectively. Then a K-step reasoning is performed on these representations to predict the final label. The initial state is given by $\mathbf{s} = \sum_j \alpha_j \mathbf{u_j}$ where $\alpha_j = \frac{\exp(\mathbf{w}^T \mathbf{u_j})}{\sum_i \exp(\mathbf{w_1}^T \mathbf{u_i})}$. On subsequent iterations $k \in [1, K-1]$, the state is $\mathbf{s}^k = GRU(\mathbf{s}^{k-1}, \mathbf{x}^k)$ where $\mathbf{x}_k = \sum_j \beta_j \mathbf{v_j}$ and $\beta_j = softmax(\mathbf{s}_{k-1} \mathbf{w_2}^T \mathbf{v})$. Then a single-layer classifier predicts the label at each iteration k:

$$P^k = softmax(\mathbf{w_3}^T [\mathbf{s}^k; \mathbf{x}^k; |\mathbf{s}^k - \mathbf{x}^k|; \mathbf{s^k}.\mathbf{x^k}])$$

Finally, all the scores across the K iterations are averaged for the final prediction. We now describe the modifications made to this model for each respective task.

2.3 Natural Language Inference

This task consists of identifying three inference relations between two sentences: Entailment, Neutral and Contradiction

Data: The data is based off the MedNLI dataset introduced by Romanov and Shivade (2018). The statistics of the dataset can be seen in Table 1.

Data Pre-Processing: On manual inspection of the data, we observe the presence of abbreviations in the premise and hypothesis. Since lexical overlap is a strong indicator of entailment by virtue of pre-trained embeddings on large corpora, the presence of abbreviations makes it challenging. Therefore, we expand the abbreviations using the following two strategies:

1. *Local Context:* We observe that often an abbreviation is composed of the first letters of contiguous words. Therefore, we first construct potential abbreviations by concatenating first letter of all words in an sequence, after tokenization. For instance, for the premise shown below we get {CXR, CXRS, XRS, CXRSI, XRSI, RSI, etc}. This is done for both the premise and the hypothesis. We then check if this n-gram exists in the hypothesis (or the premise). If yes, then we replace that abbreviation with all the words that make up the n-gram. Now the model has more scope of matching two strings lexically. We demonstrate an example below:

   Premise: Her **CXR** was clear and it did not appear she had an infection.
   Hypothesis: **Chest X-Ray** showed infiltrates.

   Premise Modified: Her **Chest X-Ray** was clear and it did not appear she had an infection.

2. *Gazetteer:* If either the premise/hypothesis does not contain the abbreviation expansion or contains only partial expansion, the *Local Context* technique will fail to expand those abbreviations. Hence, we use an external gazetteer extracted from CAMC[1] to expand commonly occurring medical terms. There were 1373 entries in the gazetteer, covering common medical and clinical expansions. For instance,
   Premise: On arrival to the **MICU** , patient is hemodynamically stable .
   Premise Modified: On arrival to the **Medical Intensive Care Unit** , patient is hemodynamically stable .

We first performed the local context replacement as they are more specific to a given premise-hypothesis pair. If there was no local context match, then we did a gazetteer lookup. It is to be noted that one abbreviation can have multiple expansions in the gazatteer and thus we hypothesized that local context should get preference while expanding the abbreviation.

Training Procedure: For training the MT-DNN model, we use the same hyper-parameters provided by the authors (Liu et al., 2019). We train model for 4 epochs and early stop when the model reaches the highest validation accuracy.

Baselines: We use the following baselines similar to Romanov and Shivade (2018).

- *CBOW:* We use a Continuous-Bag-Of-Words (CBOW) model as our first baseline. We take both the premise and the hypothesis and sum the word embeddings of the respective statements to form the input layer to our CBOW model. We used 2 hidden layers and used softmax as the decision layer.

- *Infersent:* Inferesent is a sentence encoding model which encodes a sentence by doing a max-pool on all the hidden states of the LSTM across time steps. We follow the authors of Romanov and Shivade (2018) by using shared weights LSTM cell to get the sentence representation of the premise(U) and the hypothesis(V). We feed these representations U and V to an MLP to perform a 3 way prediction. For our experiments, we use the pre-trained embeddings trained on the MIMIC dataset by Romanov and Shivade (2018). We used the same hyperparameters.

- *BERT:* Since MT-DNN is based off of the BERT (Devlin et al., 2018) model as the encoder, we also compare results using just the pre-trained BERT. We used *bert-base-uncased* model which was trained for 3 epochs with a learning rate of 2e-5 and a batch size of 16 with a maximum sequence length of 128. WE used the last 12 pre-trained layers of the model.

2.3.1 Results and Discussion

In this section we discuss the results of all of our experiments on the NLI task.

Ablation Study: First, we conduct an ablation study to study the effect of abbreviation expansion. Table 2 shows the results of the two abbreviation expansion techniques for the Infersent model. We observe the best performance with the *Gazetteer* strategy. This is because most sentences in the dataset did not have the abbreviation

[1] https://www.camc.org/

Model Ablation	Accuracy
Infersent	78.8 +/- 0.06
Infersent + Local-Context	78.8 +/- 0.02
Infersent + Local-Context + Gazetteer	78.5 +/- 0.36
Infersent + Gazetteer	**79.1 +/ 0.14**

Table 2: The results reported in the table is mean and variance of the models averaged on 3 runs using different random seeds.

matched through the local context match. Since expanding abbreviations helped increase lexical overlap, going forward we use the expanded abbreviation data for all our experiments henceforth. Table 3 shows the confusion matrix for the Infersent model. The rows represent the ground truth and the columns represent the predictions made by us. We can see that the model is most confused about the *entailment* and *neutral* classes. 82 times the model predicts *neutral* for *entailment* and 85 times vice versa. In order to address this issue, we add a prior on the dataset as a post processing step.

	Contradiction	Entailment	Neutral
Contradiction	396	43	26
Entailment	30	353	**82**
Neutral	23	**85**	357

Table 3: Confusion matrix for NLI classes for Infersent model. Rows denote the true labels and columns denote the model predictions.

Prior on the dataset: Our dataset analysis on the validation set revealed that there were three hypothesis for a given premise with mutually exclusive labels. Since we know that for a given premise there can only be one entailment because of the nature of the dataset, we post-process the model predictions to add this constraint. For each premise we collect the prediction probability for each of the hypothesis and pick the hypothesis having the highest probability for entailment. We perform the same selectional preference procedure on the remaining two classes. Such a post-processing ensures that each premise always has three hypotheses with mutually exclusive labels.

Table 4 documents the results of the different models on the validation set. We observe that our method gives the best performance among the three baselines. Based on these results, our final submission on the unseen data can be seen in the last row.

Model Ablation	Accuracy
CBOW	74.7
Infersent	79.1
BERT	80.4
Ours	**82.1**
(Ben Abacha et al., 2019) (Unseen Test)	71.4
Ours (Unseen Test)	79.6
Ours (Unseen Test) + Prior	**85.5**

Table 4: NLI results on the validation set.

2.3.2 Error Analysis

We perform qualitative analysis of our model and bucket the errors into the following categories.

1. **Lexical Overlap:** From Table 6, we see that there is a high lexical overlap between the premise and hypothesis, prompting our model to falsely predict *entailment*.

2. **Disease-Symptom relation:** In the second example, we can see that our model lacks sufficient domain knowledge to relate *hyperglycemia* (a symptom) to *diabetes* (a disease). The model interprets these to be two unrelated entities and labels as *neutral*.

3. **Drug-disease relation:** In the final example we see that our model doesn't detect that the drug names in the premise actually entail the condition in hypothesis.

These examples show that NLI in the medical domain is very challenging and requires integration of domain knowledge with respect to understanding complex drug-disease or symptom-disease relations.

2.4 Recognizing Question Entailment

This task focuses on identifying entailment between two questions and is referred as recognizing question entailment (RQE). The task is defined as : ”a question A entails a question B if every answer to B is also a complete or partial answer to A”. One of the questions is called CHQ and the other FAQ.

Data: The data is based on the RQE dataset collected by Abacha and Dina (2016). The dataset statistics can be seen in Table 7.

Pre-Processing: Similar to the NLI task, we pre-process the data to expand any abbreviations in the CHQ and FAQ.

Type	CHQ	FAQ	Label
Train	What is the treatment for tri-iodothyronine thyrotoxicosis?	What is the treatment for T3 (triiodothyronine) thyrotoxicosis?	True
	Do Coumadin and Augmentin interact?	How do you inject the bicipital tendon?	False
Validation	sepsis. Can sepsis be prevented. Can someone get this from a hospital?	Who gets sepsis?	True
	medicine and allied. I LIKE TO KNOW RECENT THERAPY ON ARRHYTHMIA OF HEART	What is an Arrhythmia?	False

Table 5: Examples of question entailment from train and validation set.

Lexical Overlap	**Premise**	She is on a low fat diet
	Hypothesis	She said they also have her on a low salt diet.
	Ground truth	Neutral
	Prediction	Entailment
Disease-Symptom relation	**Premise**	Patient has diabetes
	Hypothesis	The patient presented with a change in mental status and hyperglycemia.
	Ground truth	Entailment
	Prediction	Neutral
Drug-Disease relation	**Premise**	She was treated with Magnesium Sulfate, Labetalol, Hydralazine and bedrest as well as betamethasone.
	Hypothesis	The patient is pregnant
	Ground truth	Entailment
	Prediction	Neutral

Table 6: Qualitative analysis of the outputs produced by our model. We categorize the errors into different buckets and provide cherry-picked examples to demonstrate each category.

Label	Train Set	Validation set
True	4655	129
False	3933	173

Table 7: The number of train and validation instances in each of the categories of the RQE dataset.

Training Procedure: The multi-task MT-DNN model gave the best performance for the NLI task, which motivated us to use it for the RQE task as well. We use the same hyperparamters as Liu et al. (2019) and train the model for 3 epochs.

Baselines: We compare our model with the following baselines:

- *SVM:* Similar to Abacha and Dina (2016), we use a feature based model SVM and Logistic Regression for the task of question entailment. We extract the features presented in Abacha and Dina (2016) to the best of our abilities. Their model uses lexical features such as word overlap, bigram proportion, Named Entity Recognition (NER) features and features from the Unified Medical Concepts (UMLS) repository. Due to access issues, we only use the i2b2 [2] corpus for extracting the NER features.
- *BERT:* Like before, we compare our model with the pre-trained BERT model. For this task, we used the *bert-base-uncased* model and fine-tuned the last 12 layers for 4 epochs with learning rate 2e-5. A batch size of 16 was used.

2.4.1 Distribution Mismatch Challenges

The RQE dataset posed many unique challenges, the main challenge being that of distribution mismatch between the train and validation distribution. Table 5 shows some examples from the training and validation set which illustrate these challenges. We observe that in the training set, entailing examples always have high lexical overlap. There were about 1543 datapoints in the training set where the CHQ and FAQ were exact duplicates. The non-entailing examples in the training

[2] https://www.i2b2.org/NLP/DataSets/

set are completely un-related and hence the negative examples are not strong examples. Whereas in the validation set the negative examples also have lexical overlap. Furthermore, the nature of text in the validation set is more informal with inconsistent casing, punctuation and spellings whereas the training set is more structured. Furthermore, the length of the CHQ in the validation set is much longer than those observed in the training set. Therefore, we design our experimental settings based on these observations.

2.4.2 Data Augmentation

In order to address these challenges, we attempt to create synthetic data which is similar to our validation set. Another motivation for data augmentation was to increase the training size because neural networks are data hungry. Since most deep neural models rely on lexical overlap as strong indicator of entailment, we therefore use the UMLS features to augment our training set, but such that they help disambiguate the false positives. We use the following procedure for data augmentation:

1. We retrieve UMLS features for each question in the training, validation and test datasets, using the MetaMap [3] classifier.

2. We use the retrieved concept types and canonical names to create a new question-pair with the same label as shown in Figure 2, where the phrase *primary ciliary dyskinesia* has been replaced by its canonical name *kartaganer syndrome* and concept type *Disease or Syndrome*. Since BERT and MT-DNN have been trained on vast amount of English data including Wikipedia, the models are sensitive to language structure. Therefore, while augmenting data with UMLS features, we attempt to maintain the language structure, as demonstrated in Figure 2. Since UMLS provides the canonical features for each phrase in the sentence, we replace the found phrase with the following template < *UMLS Canonical name >, a <UMLS Concept Type>.*

Along with the synthetic data, we also experiment with another question entailment dataset Quora-Question Pairs (QQP). We describe the different training data used in our experiments:

1. *Orig:* Using only the provided training data.

2. *DataAug:* Using the validation set augmented with the UMLS features as discussed above. The provided training data was not used in this setting because of distribution mismatch. Despite the validation set being low-resources (300 sentences), MT-DNN has shown the capability of domain adaptation even in low-resource settings.

3. *QQP:* Quora Question pair [4](QQP) is a dataset which was released to identify duplicate questions on Quora. Questions are considered duplicates if the answer to one question can be be used as the answer to another question. We hypothesized that jointly training the model with the Quora-Question Pairs dataset should help as it is closest to our RQE dataset in terms of online forum data. We choose a subset of approx. 9k data points from QQP as this dataset has 400k training data points, in order to match the data points from the RQE training data. Along with this we use the validation set to train our model.

4. *Paraphrase:* Generated paraphrases of the *DataAug* using an off-the-shelf tool [5]. This was inspired by the observation that validation set was in-domain but since it was low-resourced, this tool provides a cheap way of creating additional artificial dataset.

2.4.3 Results and Discussion

The results over the validation set are in Table 9. We see that the MT-DNN model performs the best amongst all the other models. Addition of the *QQP* datasets did not add extra value. We hypothesize that this is due to lack of in-domain medical data in the QQP dataset.

The results of the MT-DNN model with the different training settings can be seen in Table 10. The test set comprises of 230 question pairs. We observe that the *DataAug* setting where the MT-DNN model is trained on in-domain validation set augmented with UMLS features, performs the best amongst all the strategies. Similar to the validation set, in this setting we also modify the test set with the UMLS features by augmenting it using the procedure of data augmentation described

[3] https://metamap.nlm.nih.gov

[4] https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs

[5] https://paraphrasing-tool.com

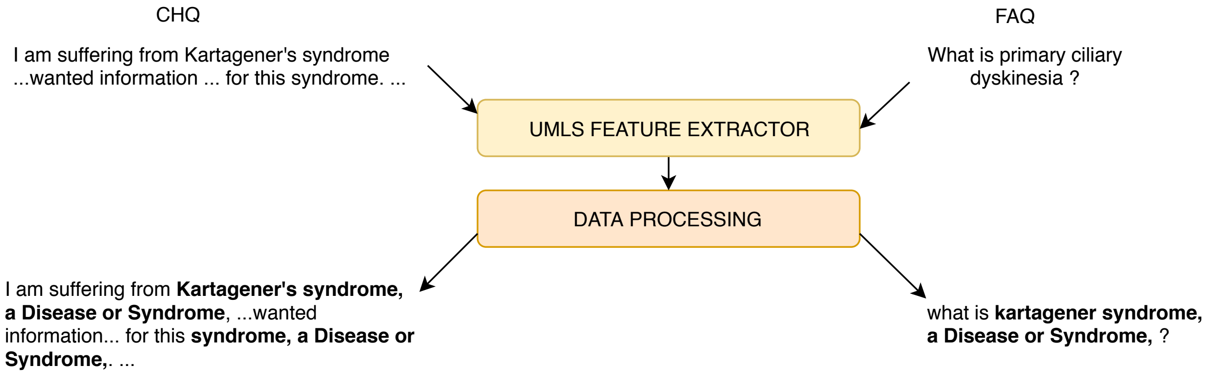
CHQ
I am suffering from Kartagener's syndrome ...wanted information ... for this syndrome. ...
FAQ
What is primary ciliary dyskinesia ?
UMLS FEATURE EXTRACTOR
DATA PROCESSING
I am suffering from **Kartagener's syndrome, a Disease or Syndrome**, ...wanted information... for this **syndrome, a Disease or Syndrome**,. ...
what is **kartagener syndrome, a Disease or Syndrome,** ?

Figure 2: Data augmentation using domain knowledge for RQE.

Lexical Overlap	**CHQ**	Please i want to know the cure to Adenomyosis... I want to see a specialist doctor to help me out.
	FAQ	Do I need to see a doctor for Adenomyosis ?
	Ground truth	False
	Prediction	True
Multiple Questions	**CHQ**	Bipolar and Generalized Anxiety Disorder I read about Transcranial magnetic stimulation Therapy. Do you know anything about it? Has it had success? Also wondering about ECT? ... Is that true for mixed bipolar and generalized anxiety disorder along with meds? Have you ever heard of this?
	FAQ	How effective is Transcranial magnetic stimulation for GAD?
	Ground truth	True
	Prediction	False
Co-reference	**CHQ**	spina bifida; vertbral fusion;syrinx tethered cord. can u help for treatment of these problem.
	FAQ	Does Spina Bifida cause vertebral fusion?
	Ground truth	True
	Prediction	True

Table 8: Qualitative analysis of the outputs produced by our RQE model. We categorize the errors into different buckets and provide cherry-picked examples to prove our claim.

above. Therefore, the test set now comprises of 460 question pairs. We refer to the provided test set of 230 pairs as *original* and the augmented test set as *UMLS*. We submitted the outputs on both the original test set and the UMLS augmented test set and observe that the latter gives **+4.3** F1 gain over the original test set. We hypothesize that the addition of the UMLS augmented data in the training process helped the model to disambiguate false negatives.

Despite training data being about medical questions, it has a different data distribution and language structure. Adding it actually harms the model, as seen by the *+ Orig + DataAug + QQP* model. For our final submission, we took an ensemble of all submissions using a majority vote strategy. The ensemble model gave us the best performance.

Model	**Accuracy**	**F1**
Abacha and Dina (2016)	-	75.0
SVM	71.9	70.0
BERT	76.2	76.2
MT-DNN + Orig	78.1	**77.4**
MT-DNN + QQP	**80.8**	77.2

Table 9: Results on the RQE validation set.

	Model	**F1**
	Ben Abacha et al. (2019)	54.1
MT-DNN	+ Orig	58.9
	+ Orig + DataAug + QQP	60.6
	+ DataAug (UMLS)	**64.9**
	+ DataAug (original)	61.5
	+ DataAug + QQP (UMLS)	**64.9**
	Ensemble	**65.8**

Table 10: Results on the RQE test set.

	Questions	Avg answer count	Avg answer length
Train set 1	104	8	434.8
Train set 2	104	8	432.5
Validation set	25	9	420.4
Test set	150	7	418.0

Table 11: Dataset statistics for re-ranking task.

2.4.4 Error Analysis

Since we used the validation set for training the model, we cannot directly perform a standard error analysis. However, we manually analyze 100 question pairs from the test set and look at the different model predictions. We categorize errors into the following categories, as shown in Table 8.

1. **Lexical Overlap:** Most of the models we used above rely strongly on lexical overlap of tokens. Therefore, question-pairs with high orthography overlap have a strong prior for the *True* label denoting entailment.

2. **Multiple-Questions:** Often CHQ questions contained multiple sub-questions. We hypothesize that multiple questions tend to confuse the model. Furthermore, as seen in Table 8, the FAQ entails from two sub-questions in the CHQ. This shows that the model lacks the ability to perform multi-hop reasoning.

3. **Co-reference:** The model is required to perform entity co-reference as part of the entailment. In the example shown in Table 8, majority of our models marked this as entailment purely because of lexical overlap. However, there was a need for the model to identify co-reference between *these problem* and the problems mentioned in the previous sentence.

2.5 Question-Answering

In this section, we focus on building a re-ranker for question-answering systems. In particular, we attempt to use the NLI and RQE models for this task. In the ACL MediQA challenge, the question-answering system CHiQA [6] provides a possible set of answers and the task is to rank them in the order of relevance.

Data: The task-3 dataset comprises of 2 training sets and a validation set. The distribution of the data across train, validation and test was consistent in terms of average number of answer candidates and average answer length per questio can be seen in Table 11.

2.5.1 Our Method

We implement the following re-ranking methods.

BM25: This is a ranking algorithm used for relevance based ranking given query. The formulation is given below:

$$\operatorname{score}(D, Q)=\sum_{i=1}^{n} \operatorname{IDF}\left(q_{i}\right) \cdot \frac{f\left(q_{i}, D\right) \cdot\left(k_{1}+1\right)}{f\left(q_{i}, D\right)+k_{1} \cdot\left(1-b+b \cdot \frac{|D|}{\operatorname{avgd}}\right)} \tag{1}$$

$$\operatorname{IDF}\left(q_{i}\right)=\log \frac{N-n\left(q_{i}\right)+0.5}{n\left(q_{i}\right)+0.5} \tag{2}$$

Here D is the answer. Q is a list of all words in the question. q_i refers to a single word. $f(q_i, D)$ is the term frequency of q_i in document D. avgd is the average answer length. The hyper-parameters used for this experiment were b = 0.75 and k1= 1.2. As shown in table 12 this gave an accuracy of 66.6 on the validation set.

NLI-RQE based model: In our second approach we leverage the pre-built NLI and RQE models from Task 1 and 2 by including the NLI and RQE scores for each question-answer pair as a feature. For instance, given a question, for each answer snippet we compute NLI scores for each sentence in the answer with the question. Since the answer snippet also contains sub-questions, we use the RQE scores to compute entailment with the question. This is illustrated below:

Question: ”about uveitis. IS THE UVEITIS, AN AUTOIMMUNE DISEASE”

For the NLI scoring we would consider statements from the answer which might predict entail, contradict or neutral for the pair. Such as *Uveitis is caused by inflammatory responses inside the eye.*

Similarly we use the question phrases from the answer to give the particular answer a RQE score based on the number of entailments *Facts About Uveitis (What Causes Uveitis?)*

[6] https://chiqa.nlm.nih.gov/

Finally, we use the BM25 score for the given answer and concatenate with the above features and use SVM as the classifier.

Model	Accuracy %
BM-25	66.6
RQE+NLI+Source	**67.5**
Ben Abacha et al. (2019) (Unseen Test)	51.7
Ours	56.5

Table 12: Accuracy for task 3 on both validation set (top) and test set (bottom).

2.5.2 Results

Table 12 documents the results of our experiments. We observe that adding NLI and RQE as features show some improvement over the BM25 model.

3 Conclusion and Future Work

In this work, we present a multi-task learning approach for textual inference and question entailment tailored for the medical domain. We observe that incorporating domain knowledge for specialized domains such as the medical domain is necessary. This is because models such as BERT and MT-DNN have been pre-trained on large amounts of generic domains, leading to possible domain mismatch. In order to achieve domain adaptation, we explore techniques such as data augmentation using UMLS features, abbreviation expansion and observe a gain of +10.8 F1 for RQE. There are still many standing challenges such as incorporating common-sense knowledge apart from domain knowledge and multi-hop reasoning which pose an interesting future direction.

In the future, we also plan to explore other ranking methods based on relevancy feedback or priority ranking for task 3. We believe using MedQuad (Ben Abacha and Demner-Fushman, 2019) as training set could further help improve the performance.

Acknowledgement

We are thankful to the anonymous reviewers for their valuable suggestions.

References

Asma Ben Abacha and Demner-Fushman Dina. 2016. Recognizing question entailment for medical question answering. In *AMIA Annual Symposium Proceedings*, volume 2016, page 310. American Medical Informatics Association.

Asma Ben Abacha and Dina Demner-Fushman. 2019. A question-entailment approach to question answering. *arXiv e-prints*.

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.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Carol Friedman, Pauline Kra, and Andrey Rzhetsky. 2002. Two biomedical sublanguages: a description based on the theories of zellig harris. *Journal of biomedical informatics*, 35(4):222–235.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2018. Scitail: A textual entailment dataset from science question answering. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Alexey Romanov and Chaitanya Shivade. 2018. Lessons from natural language inference in the clinical domain. *arXiv preprint arXiv:1808.06752*.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.