ChicHealth @ MEDIQA 2021: Exploring the limits of pre-trained seq2seq models for medical summarization

Liwen Xu, Yan Zhang, Lei Hong, Yi Cai*, Szui Sung
Chic Health, Shanghai, China

Abstract

In this article, we will describe our system for MEDIQA2021 shared tasks. First, we will describe the method of the second task, multiple answer summary (MAS). For extracting abstracts, we follow the rules of Xu and Lapata (2020). First, the candidate sentences are roughly estimated by using the Roberta model. Then the Markov chain model is used to evaluate the sentences in a fine-grained manner. Our team won the first place in overall performance, with the fourth place in MAS task, the seventh place in RRS task and the eleventh place in QS task. For the QS and RRS tasks, we investigate the performanceS of the end-to-end pre-trained seq2seq model. Experiments show that the methods of adversarial training and reverse translation are beneficial to improve the fine tuning performance.

1 Introduction

The mediqa 2021 shared tasks aim to investigate the most advanced summary models, especially their performance in the medical field. There are three tasks. The first is question summary (QS), which classifies long and complex consumer health problems into simple ones, which has been proved to be helpful to answer questions automatically (Abacha and Demner-Fushman, 2019). The second task is multiple answer summary (MAS) (Savery et al., 2020). Different answers can bring complementary views, which may benefit the users of QA system. The goal of this task is to develop a system that can aggregate and summarize answers scattered across multiple documents. The third task is radiology report summary (RRs) (Zhang et al., 2018, 2020b), which generates radiology impression statements by summarizing the text results written by radiologists.

Automatic summarization is an important task in the field of medicine. When users use Google, MEDLINE and other search engines, they need to read a large number of medical documents about a certain topic and get a list of possible answers, which is very time-consuming. First, the content may be too specialized for laymen to understand. Second, one document may not be able to fully answer queries, and users may need to summarize conclusions across multiple documents, which may lead to a waste of time or misunderstanding. In order to improve the user experience when using medical applications, automatic summarization technology is needed.

In the MAS task, we improve upon (Xu and Lapata, 2020) via three methods. First, during the coarse ranking of a sentence in one of the given documents, we also add the surrounding sentences as input and use two special tokens marking the positions of the sentence. This modification improves the coarse ranking with a large margin. Second, due to the low resource settings of this task, we find that applying a RoBERTa (Liu et al., 2019) model which is already fine-tuned on the GLUE benchmark (Wang et al., 2018) can be beneficial.

In the MAS task, we use two methods to improve (**?**). First, when we rank a sentence coarsely in a given document, we add the surrounding sentences as input. This modification greatly improves the efficiency of coarse ranking. Secondly, due to the low resource setting of this task, we find that it is beneficial to apply the Roberta (Liu2019RoBERTaAR) model, which has been fine tuned on the glue benchmark (Wang2018GLUEAM).

For the other two tasks, we mainly discuss how the pre trained seq2seq model, such as Bart (Lewis et al., 2020), Pegasus (Zhang et al., 2020a), can be implemented in these tasks. You can make two takeout. First, for tasks with smaller datasets, freezing part of the parameters is beneficial. Second, backtranslation is beneficial for generalization.

Our team ChicHealth participated in all three tasks and won the first place for the overall per-

*Contact author: yi.cai@chic-health.com.

Proceedings of the BioNLP 2021 workshop, pages 263–267

formances. Experiments show that our methods are beneficial for pre-trained models' downstream performances.

2 Extractive MDS

Let Q denote a query, and $D = \{d_1, d_2, ..., d_M\}$ a set of documents. We have implemented multi granularity MDS following the implementation of Xu and Lapata (2020). We first break down the document into paragraphs, which are sentences. Then, a trained Roberta model quantifies the semantic similarity between the selected sentence and the query, and estimates the importance of the sentence (evidence estimator) according to the sentence itself or the local context of the sentence. Thirdly, in order to give the global estimation of the importance of each part in the summary, we use the centrality estimator based on the Markov chain.

2.1 Evidence Estimator

Let $\{S_1, S_2, ..., S_N\}$ as the candidate answer set. Our training goal is to find the right answers in this group. We use Roberta as our sequence encoder

We concatenate query Q after candidate sentence S into a sequence $< /s >, S, < /s >< s >, Q, < /s >$, as the input to the RoBERTa encoder. The starting $< s >$ token's vector representations t serves as input to a single layer feed forward layer to obtain the distribution over positive and negative classes, where the positive class denotes that a sentence contains the answer and 0 otherwise.

We connect the query Q to the sequence $< s >, S, < /s >, Q, < /s >$ after the candidate statement $s$as the input of the Roberta encoder. The vector of the starting $< s >$ is used as the input of the single feed-forward layer to obtain the distribution on the positive and negative classes, where the positive class indicates that the sentence contains the answer, otherwise it is 0. We can improve the performance of the evidence estimator by adding the surrounding sentences of S into the model during training.

After fine-tuning, we take the probability of positive class as the score of local evidence, and we will use it to sort all sentences of each query.

2.2 Centrality Estimator

In order to obtain a global estimate of the score of each candidate sentence, we apply a global estimator following Xu and Lapata (2020). The centrality estimator is essentially an extension of the famous LexRank algorithm (Erkan and Radev, 2004).

For each document cluster, i.e., the collections of documents for each query in our tasks, LexRank builds a graph $G = (V; E)$ with nodes V corresponding to sentences and undirected edges E whose weights are computed based on a certian similarity metric. The original LEXRANK algorithm uses TF-IDF (Term Frequency Inverse Document Frequency). (Xu and Lapata, 2020) proposes to use TF-ISF (Term Frequency Inverse Sentence Frequency), which is similar to TF-IDF but operates at the sentence level.

Following ((Xu and Lapata, 2020)), the similarity matrix E is combined with the evidence estimator's , that is,

$$\tilde{E} = w * [\tilde{q}; ...; \tilde{q}] + (1 - w) * E, \quad (1)$$

where $w \in (0, 1)$ controls the extent to which the evidence estimator can influence the final summarization, and $\tilde{q}$ is obtained by normalizing the the evidence scores,

$$\tilde{q} = \frac{q}{\sum_v^{|V|} q_v}. \quad (2)$$

We run a Markov Chain on the graph and the final stationary distribution $\tilde{q}^*$ of this Markov chain serves as the final scores of each sentence.

3 Abstractive summarization

Pre-trained models. In this section, we investigate the pretrained Seq2Seq models to obtain abstractive summarizations, after finetuning their on our datasets. We mainly investigate two types of models, BART ((Lewis et al., 2020)) and PEGASUS ((Zhang et al., 2020a)). And experiments show the PEGASUS model is better

Finetuning techniques. In order to fine tune the pre-trained seq2seq model, we test some methods/techniques that can improve the performance of downstream tasks:

- Freezing a proportion of the parameters of the model;
- Advarsarial training method, i.e., Projected Gradient Descent (PGD, (Madry et al., 2018)).
- Backtraslation (from English to Thai, and then Thai to English) is applied for data augmentation.

model	ROUGE-2 F1
BART-base	11.47
BART-large	13.73
PEGASUS-large	16.37

Table 1: Comparison of different pretrained models on valid set in Task 1.

model	# layers to freeze	ROUGE-2
PEGASUS-large	3	16.37
PEGASUS-large	0	15.80
PEGASUS-large	6	14.98
PEGASUS-large	9	15.64
PEGASUS-large	12	9.85

Table 2: Results of PEGASUS-large model, when we freeze different numbers of lower layers of the encoder and decoder.

with Adv training?	ROUGE-2
Yes	16.37
No	15.46

Table 3: Results of PEGASUS-large model, with or without adversarial training.

evidence estimator	centrality estimator	ROUGE-2
dev set		
roberta-base	No	44.32
roberta-large	No	46.48
roberta-large + GLUE finetuning	No	47.13
roberta-large + GLUE finetuning	LexRank	48.24
ensemble models	LexRank	49.18

Table 4: Comparison of different models on dev set of the MAS task.

model	ROUGE-2
BERT-abs	34.95
T5-small	45.46
T5-base	49.41
T5-large	50.68
BART-base	49.65
BART-large	49.81
PEGASUS-pubmed	45.93
PEGASUS-large	**51.95**

Table 5: The results of different summarization models.

4 Experiments

4.1 dataset statistics

For QS tasks (Figure 1 and 2), the source length distribution is consistent on the train/Val/test set, and the target length distribution is also consistent. For RRS tasks (7 and 8), we can observe that the sequence length distribution of train/ val/test set is different, which may lead to skewed model. For task 2, the length of the document varies, which is too long for pre-trained models like Pegasus. Therefore, for task 2, abstractive summaries are generated from extractive summaries.

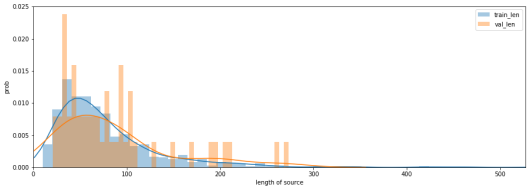

Figure 1: Source sequence length of QS.

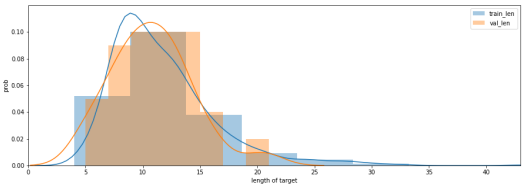

Figure 2: Target sequence length of QS.

4.2 Results on QS

We first report the results on the QS task. First, we compare BART and PEGASUS (Table 1), and find that PEGASUS performs significantly better than BART. Second, we compare PEGASUS with different number of layers freezed (Table 2), and find that freezing three 3 layers obtains the best dev performance. Third, we compare the model with or without adversarial training (Table 3), and show that adversarial training is important for this task.

4.3 Results on MAS

Now we report results on the MAS task (Table 4). RoBERTa large performs better on coarse ranking than RoBERTa base. And using a model finetuned on GLUE also helps to improve the fine-tuning task. After centrality ranking with LexRank, the score improve by more than one percent. And our best score is obtained by using ensemble on the evidence estimators.

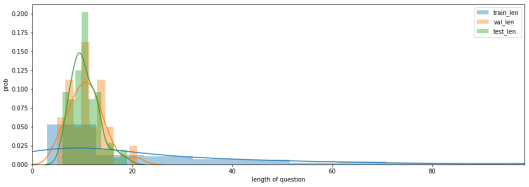

Figure 3: Query length of MAS.

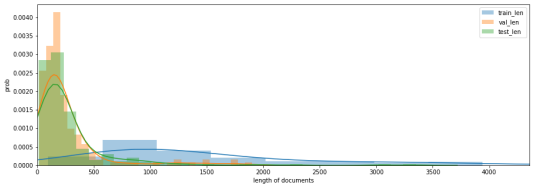

Figure 4: Document length of MAS.

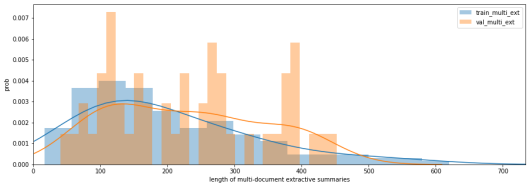

Figure 5: Extractive summary length of MAS.

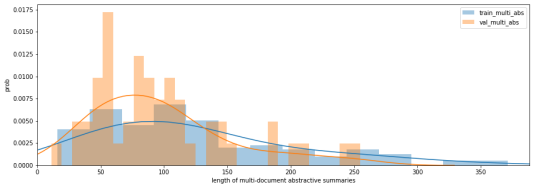

Figure 6: Abstractive summary length of MAS.

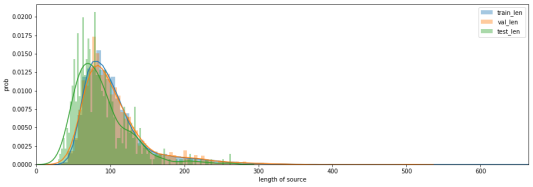

Figure 7: source length of task3 using PEGASUS tokenizer

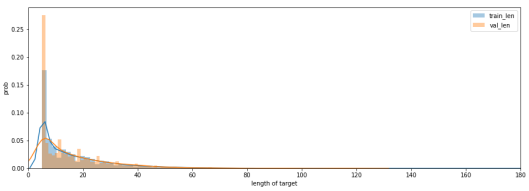

Figure 8: target length of task2 using PEGASUS tokenizer

4.4 Results on RRS

Now we report results on the RRS task. We compare 4 groups of models, BERT-abs, T5 (Raffel et al., 2020), BART and PEGASUS (Table 5). PEGASUS also performs best, like in the QS task. However, we find that the PEGASUS trained on PubMed performs significant worse, which is contradictory to our hypothesis that fine-tuning on related domain corpus is beneficial for downstream tasks.

5 Conclusion

In this work, we elaborate on the methods we employed for the three tasks in the MEDIQA 2021 shared tasks. For the extractive summarization of MAS task, we build upon Xu and Lapata (2020), and achieve improvements by adding contexts and sentence position markers. For generating abstractive summaries, we leverage the pre-trained seq2seq models. To improve the fine-tuning performances on the downstream tasks, we implement a few techniques, like freezing part of the models, adversarial training and back-translation. Our team achieves the 1st place for the overall performances.

In this work, we elaborate the methods used in the three shared tasks of mediqa 2021. For MAS task, we employ the methods that are similar to Xu and Lapata (2020). In order to generate abstract abstracts, we take advantages of the pre-trained seq2seq model. In order to improve the fine-tuning performance of downstream tasks, we use freezing part of the model, adversarial training. Our team ranks first in the overall performances of the three task.

References

Asma Ben Abacha and Dina Demner-Fushman. 2019. On the role of question summarization and information source restriction in consumer health question answering. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science*, 2019:117–126.

Günes Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Intell. Res.*, 22:457–479.

M. Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, A. Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *ArXiv*, abs/1910.13461.

Y. Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

A. Madry, Aleksandar Makelov, L. Schmidt, D. Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. *ArXiv*, abs/1706.06083.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, W. Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Max E. Savery, Asma Ben Abacha, Soumya Gayen, and Dina Demner-Fushman. 2020. Question-driven summarization of answers to consumer health questions. *Scientific Data*, 7.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *BlackboxNLP@EMNLP*.

Yumo Xu and Mirella Lapata. 2020. Coarse-to-fine query focused multi-document summarization. In *EMNLP*.

Jingqing Zhang, Y. Zhao, Mohammad Saleh, and Peter J. Liu. 2020a. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *ICML*.

Yuhao Zhang, D. Ding, Tianpei Qian, Christopher D. Manning, and C. Langlotz. 2018. Learning to summarize radiology findings. In *Louhi@EMNLP*.

Yuhao Zhang, Derek Merck, E. Tsai, Christopher D. Manning, and C. Langlotz. 2020b. Optimizing the factual correctness of a summary: A study of summarizing radiology reports. In *ACL*.