SB_NITK at MEDIQA 2021: Leveraging Transfer Learning for Question Summarization in Medical Domain

Spandana Balumuri, Sony Bachina and Sowmya Kamath S
Healthcare Analytics and Language Engineering (HALE) Lab,
Department of Information Technology,
National Institute of Technology Karnataka, Surathkal, Mangalore 575025, India
{spandanabalumuri99, bachina.sony}@gmail.com
sowmyakamath@nitk.edu.in

Abstract

Recent strides in the healthcare domain, have resulted in vast quantities of streaming data available for use for building intelligent knowledge-based applications. However, the challenges introduced to the huge volume, velocity of generation, variety and variability of this medical data have to be adequately addressed. In this paper, we describe the model and results for our submission at MEDIQA 2021 Question Summarization shared task. In order to improve the performance of summarization of consumer health questions, our method explores the use of transfer learning to utilize the knowledge of NLP transformers like BART, T5 and PEGASUS. The proposed models utilize the knowledge of pre-trained NLP transformers to achieve improved results when compared to conventional deep learning models such as LSTM, RNN etc. Our team SB_NITK ranked 12th among the total 22 submissions in the official final rankings. Our BART based model achieved a ROUGE-2 F1 score of 0.139.

1 Introduction

The Question Summarization (QS) task aims to promote the development of new summarization models that are able to summarize lengthy and complex consumer health questions. The consumer health questions can have a variety of subjects like medications, diseases, effects, medical treatments and procedures. The medical questions can also contain a lot of irrelevant information that makes automated question summarization a difficult and challenging task (Mayya et al., 2021). It is also often cumbersome to go through lengthy questions during the question answering process and then formulate relevant answers (Upadhya et al., 2019). The automated summarization approaches for consumer health questions thus have many medical applications. An effective automated summarization approach for obtaining simplified medical health questions can be crucial to improving medical question answering systems.

The MEDIQA 2021 (Ben Abacha et al., 2021) proposes three different shared tasks to promote the development, performance improvement and evaluation of text summarization models in the medical domain:

- *Consumer Health Question Summarization (QS)* - Development of summarization models to produce the shortened form of consumer health related questions.
- *Multi-Answer Summarization* - Development of summarization models to aggregate and summarize multiple answers to a medical question.
- *Radiology Report Summarization* - Development of summarization models that can produce radiology impression statements by summarising text-based observations.

The role of question summarization or simplification in answering consumer health questions is not explored extensively when compared to the summarization of documents and news articles (George et al., 2021). Ishigaki et al. (2017) explored various extractive and abstractive methods for summarization of questions that are posted on a community question answering site. The results showed that abstractive methods with copying mechanism performed better than extractive methods. Agrawal et al. (2019) proposed a closed-domain Question Answering technique that uses Bi-directional LSTMs trained on the SquAD dataset to determine relevant ranks of answers for a given question. Ben Abacha and Demner-Fushman (2019) proposed sequence-to-sequence attention models with pointer generator network for summarization of consumer health questions collected from MeQSum, Quora question pairs dataset and other sources. The addition of pointer generator and cov-

Proceedings of the BioNLP 2021 workshop, pages 273–279

erage mechanisms on the sequence-to-sequence has improved the ROUGE scores considerably.

In this paper, we describe the different models and experiments that we designed and evaluated for the Consumer Health Question Summarization (QS) task. The proposed models utilize the knowledge of pre-trained NLP transformers to achieve improved results when compared to conventional deep learning models such as LSTM, RNN etc. The proposed models are based on transfer learning and fine tuning the dataset on different versions of NLP transformers like BART (Lewis et al., 2019), T5 (Raffel et al., 2020) and PEGASUS (Zhang et al., 2020). We have also benchmarked all the proposed models against traditional Seq2Seq LSTM encoder-decoder networks with attention.

The rest of this article is organized as follows. In Section 2, we provide information about the data used such as description of datasets, dataset augmentation and pre-processing. Section 3 gives an overview of transformer architecture and transfer learning. In Section 4, we describe and compare results obtained from fine-tuning various transformer models on our augmented dataset. In Section 5, we compare the performance of our proposed models with different transformer models in detail, followed by conclusion and directions for future work.

2 Data

2.1 MeQSum Dataset Description

The main dataset for the task was provided by the organizers of MEDIQA 2021 (Ben Abacha et al., 2021). The training set comprised of consumer health questions (CHQs) and the corresponding summaries. The validation set consisted of National Library of Medicine (NLM) consumer health questions and their respective summaries. In addition to the questions and summaries, the validation set contains question focus and question type for each question. The MeQSum training corpus consists of 1000 question-summary pairs while the validation dataset provided has 50 NLM question-summary pairs. To improve the performance, the question focus in validation pairs has been appended to the beginning of each question.

2.2 Dataset Augmentation

As the provided training and validation datasets for the task add up to only a 1,050 question-summary pairs, we decided to augment the data to achieve better performance and solve over-fitting problems. The following three datasets were added to the training and validation datasets to broaden the coverage.

TREC-2017 LiveQA: Medical Question Answering Task Dataset. The LiveQA dataset is used for training consumer health question answering systems. The question pairs in this dataset are very similar to those given for the task, however, its small size was not conducive to performance improvement. The test dataset (Ben Abacha et al., 2017) comprises of 104 NLM Questions, out of which 102 of them have an associated summary annotation. Additionally, each question has focus, type, and keyword annotations associated with it. To increase the weight of significant parts of the question, we added the question focus and keyword annotations to the beginning of each question.

Recognizing Question Entailment (RQE) Dataset. The RQE dataset (Ben Abacha and Demner-Fushman, 2016) is used for automatic question answering by recognizing similar questions in the medical domain. Out of the 8,588 training pairs and 302 validation pairs available in the RQE corpus, we chose only those pairs which entail each other, which resulted in 4,655 training pairs and 129 validation pairs. Moreover, to ensure that one of the questions in the pair is a summary of the other, we selected those pairs where one question has at least 2 sentences and the other has only one sentence. This resulted in a total of 2,078 question-summary pairs. However, one of the issues faced with this dataset is that the questions in some pairs are almost similar to each other.

Medical Question Pairs (MQP) Dataset. The MQP dataset (McCreery et al., 2020) consists a total of 3,048 pairs of related and unrelated medical questions. Half of the total questions i.e., 1,524 pairs are labeled as similar to each other. Among the similar question pairs, we chose those pairs where at least one of the questions has only one sentence. In case both the questions have only one sentence each, the question with lesser number of words is considered as the summary. Finally, the dataset resulted in 1,057 pairs. The advantage of MQP dataset lies in the fact that it has more generalized medical questions in contrast to the previously mentioned datasets, which have many esoteric terms.

2.3 Dataset Preprocessing

The dataset preprocessing largely depends on the data at hand and the type of output we anticipate. Some of the common techniques that we incorporated include text case-folding to lowercase, removal of special characters, numbers and stop words etc. However, upon analyzing the summaries, we found that they include uppercase letters, certain special characters, numbers and stop words. Therefore we did not proceed with extensive data preprocessing, except for removing special characters which are absent the summaries. The final cleaned corpus comprises of 4,287 question-summary pairs.

3 System Description

3.1 Transformer Architecture

Transformers have now become the state-of-the-art solution for a variety of NLP tasks including language understanding, translation, text generation, text classification, question answering and sentiment analysis. Transformers continue to outperform other neural network architectures (RNN and LSTM) by maintaining the attention while handling sequences in parallel, i.e., they handle all words at once (considered bidirectional) rather than one by one and effectively learning inter-dependencies, especially in the case of long sentences.

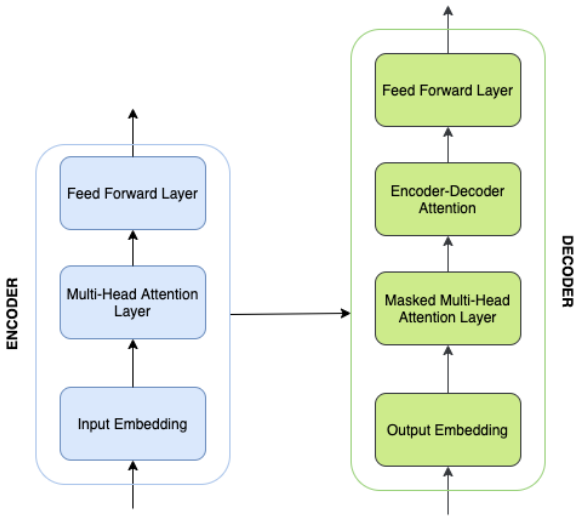

Figure 1: Encoder-Decoder transformer architecture used by PEGASUS, BART and T5.

The transformer architecture as shown in Fig. 1 consists of the encoder and decoder mechanisms, where the segments are connected by a cross-attention layer. An encoder segment consists of a stack of encoders in which each encoder reads the text input and generates embedding vectors. It outputs contextual and positional vectors of the input sequence using attention mechanism. Similarly, the decoder part is a stack of decoders where each decoder takes target sequence and encoder output as input. It generates contextual information from the target sequence and then combines encoder output with it. It models the conditional probability distribution of the target vector sequence based on the previous target vectors to produce an output vector.

The sequence of input tokens is fed into the transformer encoder, which are then embedded into vectors and processed by the neural network. The decoder produces a series of vectors, corresponding to each token in the input sequence. Few examples of existing transformers are BART, T5 etc. As deep neural networks have a large number of parameters, the majority of labelled text datasets are insufficient for training these networks as training them on limited datasets would result in over-fitting. Therefore, for downstream NLP tasks, we can utilize the knowledge of transformers which are pre-trained on large datasets using transfer learning. Transfer learning is a method of using a deep learning model that has been pre-trained on a huge corpus to perform similar NLP tasks by fine-tuning on a different dataset.

For fine-tuning the model with a different dataset, we modify the model parameters like hidden states and weights of the existing model to suit our dataset. Towards this, we have fine-tuned transformer models such as BART, T5 and PEGASUS with our augmented dataset to perform question summarization, for the given task. Fine tuning BART transformer for question summarization with our dataset achieved the best ROUGE-2 scores when compared to other transformer models. The details of experiments and analysis of different models are discussed in Section 4.

4 Models and Results

During the system development phase, we experimented with various models for the task of question summarization. The ranking for the task is based on the ROUGE2-F1 score. ROUGE-2 (Recall-Oriented Understudy for Gisting Evaluation), is a metric which measures the overlap of bigrams between the model-generated and reference summaries in a summarization task. In the following

sections, we discuss the various versions of the models that we fine-tuned for the Question Summarization task.

4.1 Seq2Seq models

This model uses a seq2seq bidirectional LSTM based encoder and decoder. The encoder network is combination of an embedding layer followed by a stack of 3 bidirectional LSTM layers each with 128 hidden units and a dropout value of 0.2. The encoder output and encoder states from the LSTM network is given as input to the attention layer (Bahdanau et al., 2016) to generate context vector and attention weights. The generated vectors from attention layer are given as input to decoder. The decoder network is similar to the encoder, having a combination of an embedding layer followed by a stack of bidirectional LSTMs of 128 hidden units and a softmax layer. The output from the decoder network is a vector of tokens' indexes from the vocabulary.

We have experimented with the following variations of seq2seq - attention - coverage model.

1. *Seq2seq + attention + coverage* model with Word2vec ($N \times 300$) embeddings.
2. *Seq2seq + attention + coverage* model with Scibert ($N \times 768$) embeddings.
3. *Seq2seq + attention + coverage* model with Glove ($N \times 300$) embeddings.

However, the above mentioned seq2seq models were not submitted for final evaluation because of the lack of sufficient data to train such models from scratch. Since the size of our training dataset is small (4,287 question-summary pairs), these seq2seq models did not provide acceptable results, hence we omitted them from our submissions for the question summarization task.

4.2 T5

Google's T5 (Text-to-Text Transfer Transformer) is a pre-trained encoder-decoder model that has been trained on C4 (Colossal Clean Crawled Corpus) dataset for unsupervised and supervised tasks. The T5 transformer consists of an encoder, a cross attention layer and an auto-regressive decoder. In T5, every NLP problem is converted to a text-to-text format and the data is augmented with a prefix e.g., for summarization: '*summarize*: ', for translation: "*translate English to French:* ". T5 achieves benchmark performance for various tasks like summarization, question answering, text classification etc, and both supervised and unsupervised methods can be applied for training. Two different versions of T5 were finetuned for our augmented dataset for the summarization task.

1. *t5-base* : T5 model with 12 encoder and decoder layers, trained on C4 dataset, with 220M parameters.
2. *t5-small* : T5 model with 6 encoder and decoder layers, trained on C4 dataset, with 60M parameters.

Table 1 shows the comparison of ROUGE scores obtained for the T5 models we experimented with. The model t5-small obtained a better ROUGE-2-F1 score when compared to t5-base. We submitted a run each for the two models. In addition to these two models, we also experimented with other variations of T5, such as *t5-large* and *t5-base-finetuned-summarize-news*. On comparison of the summaries produced by the various T5 models, t5-small generated the best summaries.

4.3 PEGASUS

Google AI released the PEGASUS model which implements the sequence-to-sequence architecture. The specialty of this model is its self-supervised pre-training objective termed as "gap-sentence generation", where, certain sentences are masked in the input for pre-training. The advantage is gained by keeping the pre-training self-supervised objective closer to the required down-stream task. We mainly focused on the following two versions of the PEGASUS models and fine-tuned them on our augmented dataset.

1. *pegasus-xsum*: pegasus-large model fine-tuned on the XSum dataset having a size of 226k records.
2. *pegasus-wikihow*: pegasus-large model fine-tuned on the WikiHow dataset having a size of 168k records.

Table 1 shows the ROUGE scores obtained for the PEGASUS models finetuned in our work. Among the two, pegasus-wikihow gives better scores than pegasus-xsum. We submitted one run for each of the models. Additionally, we also experimented with other pre-trained PEGASUS models such as, *pegasus-pubmed*, *pegasus-cnn_dailymail* and *pegasus-multi_news*. The summaries produced by these pegasus-cnn_dailymail and pegasus-multi_news were almost similar and acceptable, while those generated by pegasus-pubmed were not up to the mark.

Table 1: Scores and ROUGE values for various models benchmarked for the Question Summarization task

Model	Score	R1-P	R1-R	R1-F1	R2-P	R2-R	R2-F1	RL-R	RL-F1
bart-large-xsum	**0.139**	**0.358**	0.346	**0.333**	**0.152**	**0.144**	**0.139**	**0.318**	**0.308**
bart-large-cnn	0.12	0.339	0.299	0.301	0.137	0.117	0.12	0.274	0.276
pegasus-xsum	0.107	0.329	0.284	0.289	0.128	0.104	0.107	0.261	0.267
pegasus-wikihow	0.129	0.321	**0.349**	0.307	0.143	0.142	0.129	0.304	0.271
t5-base	0.112	0.343	0.297	0.3	0.133	0.107	0.112	0.268	0.273
t5-small	0.114	0.293	0.31	0.281	0.124	0.121	0.114	0.272	0.25

4.4 BART

BART (Bidirectional and Auto-Regressive Transformers) is based on the standard transformer architecture proposed by Facebook, having BERT (Devlin et al., 2019) like encoder and GPT (Radford et al., 2019) like decoder. The denoising objective of the encoder while the decoder that works to reproduce the original sequence, using the previously produced tokens and the encoder output, bring the best of the two models. We experimented with the following different BART pre-trained models by fine-tuning them of our augmented dataset.

1. *bart-large-xsum* : bart-large (BART with 12 encoder & decoder layers) fine-tuned on Xsum dataset with 400M parameters.

2. *bart-large-cnn* : bart-large (BART with 12 encoder & decoder layers) fine-tuned on CNN/Dailymail dataset with 400M parameters.

The ROUGE scores obtained for both the BART based models are tabulated in Table 1. The bart-large-xsum model gives a better performance than the bart-large-cnn model. We have submitted 3 runs for each of the two models, by varying the hyper-parameters such as the summary length, learning rate, length penalty and epochs. The best ROUGE scores were obtained at a learning rate of 3e-5, summary length of 30 and with no length penalty running for 3 epochs. Besides these two models, we have also experimented with other BART models, such as *bart-large-mnli* and *bart-large-gigaword*, however, the summaries generated were not at par with those of the earlier two models.

5 Comparative Evaluation

During the testing phase, we experimented with various models based on the transformer architecture, such as BART, T5 and PEGASUS as mentioned previously. We were allowed to submit a maximum of 10 runs per task. Therefore, we submitted two runs each for T5 and PEGASUS models, and six runs for various approaches of the BART model. The test set provided for the Question Summarization task comprises of 100 NLM questions with their associated question ids. The test set was pre-processed in a similar fashion as the augmented dataset we had used for training. Additionally, certain tokens such as "[NAME]", "[LOCATION]", "[CONTACT]", "[DATE]", "SUBJECT: " and "MESSAGE: " were removed from the test dataset to avoid their appearance in the generated summaries.

Table 2 shows the summaries generated by various transformer based models for a sample question in the test set. From the table it can be observed that, the summaries generated by t5-base and t5-small are almost similar and don't actually capture the main focus of the question. The summary generated by pegasus-xsum is similar but longer than those produced by the T5 models. However, the summary generated by the pegasus-wikihow model is quite apt. The bart-large-cnn model produced a summary which is although grammatically correct, the meaning is incorrect. The bart-large-xsum generated the best summary amongst all the models, because it is both precise and short in length.

The HOLMS (Mrabet and Demner-Fushman, 2020) and BERTScores (Zhang* et al., 2020) for the different models used are referenced in Table 3. Based on the experiments, it was observed that the bart-large-xsum model achieved the best performance in terms of both metrics. Based on this performance, our team ranked 2nd in the BERTScore metric and secured 6th position in HOLMS score, on the leaderboard.

Table 2: Sample summary generated by various models for the test question: *"Gadolinum toxicity and MCS relationship? I have 2 Genovia Labs test results years apart with seriously high Gadolinum toxicity. AND I am very VERY VERY very challenged by MCS - Multiple Chemical Sensitivity. My question is: If I had multiple MARs after an auto accident. And since then the MCS is debilitating. Certainly the symptoms of Gas level in my body cause symptoms as well. But I am debilitated by Synthetic chemicals in the air. How can I find out if the Gas exhaserbated my reaction to exhaust fumes, air fresheners, perfumes, dryer sheets(!!!!), food additives, and much more. Many Thanks"*

Model	Generated Summary
bart-large-xsum	What is the relationship between Gadolinum toxicity and MCS?
bart-large-cnn	What are the causes of and treatments for Multiple Chemical Sensitivity?
pegasus-xsum	How can I find out if synthetic chemicals in the air cause my reaction to exhaust fumes, air fresheners, perfumes, dryer sheets, food additives?
pegasus-wikihow	Where can I find information on Gadolinum toxicity and MCS relationship?
t5-base	How can I find out if gas exhaserbated my reaction to exhaust fumes, air fresheners, perfumes,?
t5-small	How can I find out if the Gas exhaserbated my reaction to exhaust fumes, air fresheners, perfumes?

Table 3: HOLMS and BERTScore F1 performance of the proposed models, for the Question Summarization task

Model	HOLMS	BERTScore-F1
bart-large-xsum	**0.566**	**0.702**
bart-large-cnn	0.556	0.692
pegasus-xsum	0.544	0.674
pegasus-wikihow	0.535	0.665
t5-base	0.550	0.681
t5-small	0.537	0.633

6 Conclusion and Future Work

In this paper, we presented models that explore the use of transfer learning to utilize the knowledge of NLP transformers like BART, T5 and PEGASUS for the task of question summarization. The observed scores and the sample summaries generated by different transformer architecture based models clearly delineated the best performing model among the ones proposed. The summaries produced by the bart-large-xsum achieved the best score, followed by the pegasus-wikihow model. This can be largely attributed to the transfer learning technique that was adapted, by utilizing models which are pre-trained on massive datasets. As part of future work for the question summarization task, we plan to exploit question type feature, in addition to the currently used question focus feature for further enhancing the performance.

References

Anumeha Agrawal, Rosa Anil George, Selvan Suntiha Ravi, Sowmya Kamath, and Anand Kumar. 2019. Ars_nitk at mediqa 2019: Analysing various methods for natural language inference, recognising question entailment and medical question answering system. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 533–540.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2016. Neural machine translation by jointly learning to align and translate.

Asma Ben Abacha, Eugene Agichtein, Yuval Pinter, and Dina Demner-Fushman. 2017. Overview of the medical question answering task at trec 2017 liveqa. In *TREC 2017*.

Asma Ben Abacha and Dina Demner-Fushman. 2016. Recognizing question entailment for medical question answering. In *AMIA 2016, American Medical Informatics Association Annual Symposium, Chicago, IL, USA, November 12-16, 2016*.

Asma Ben Abacha and Dina Demner-Fushman. 2019. On the summarization of consumer health questions. In *Proceedings of the 57th Annual Meeting of the*

Association for Computational Linguistics, pages 2228–2234, Florence, Italy. Association for Computational Linguistics.

Asma Ben Abacha, Yassine Mrabet, Yuhao Zhang, Chaitanya Shivade, Curtis Langlotz, and Dina Demner-Fushman. 2021. Overview of the mediqa 2021 shared task on summarization in the medical domain. In *Proceedings of the 20th SIGBioMed Workshop on Biomedical Language Processing, NAACL-BioNLP 2021*. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Rosa George, Selvan Sunitha, and S Sowmya Kamath. 2021. Benchmarking semantic, centroid, and graph-based approaches for multi-document summarization. In *Intelligent Data Engineering and Analytics*, pages 255–263. Springer.

Tatsuya Ishigaki, Hiroya Takamura, and Manabu Okumura. 2017. Summarizing lengthy questions. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 792–800, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, et al. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

Veena Mayya, Sowmya Kamath, Gokul S Krishnan, and Tushaar Gangavarapu. 2021. Multi-channel, convolutional attention based neural model for automated diagnostic coding of unstructured patient discharge summaries. *Future Generation Computer Systems*, 118:374–391.

Clara H. McCreery, Namit Katariya, Anitha Kannan, Manish Chablani, and Xavier Amatriain. 2020. Effective transfer learning for identifying similar questions: Matching user questions to covid-19 faqs.

Yassine Mrabet and Dina Demner-Fushman. 2020. HOLMS: Alternative summary evaluation with large language models. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5679–5688, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer.

Akshay Upadhya, Swastik Udupa, and S Sowmya Kamath. 2019. Deep neural network models for question classification in community question-answering forums. In *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–6. IEEE.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.